Long-term tracking and monitoring of mobile entities in the outdoors using wireless sensors

Ion Emilian Radoi

Doctor of Philosophy

Centre for Intelligent Systems and their Applications
School of Informatics
University of Edinburgh

2016

Abstract

There is an emerging class of applications that require long-term tracking and monitoring of mobile entities for characterising their contexts and behaviours using data from wireless sensors. Examples include monitoring animals in their natural habitat over the annual cycle; tracking shipping containers and their handling during transit; and monitoring air quality using sensors attached to bicycles used in public sharing schemes. All applications within this class require the acquisition of sensor data tagged with spatio-temporal information and uploaded wirelessly. Currently there is no solution targeting the entire class of applications, only point solutions focused on specific scenarios. This thesis presents a complete solution (firmware and hardware) for applications within this class that consists of attaching mobile sensor nodes to the entities for tracking and monitoring their behaviour, and deploying an infrastructure of base-stations for collecting the data wirelessly. The proposed solution is more energy efficient compared to the existing solutions that target specific scenarios, offering a longer deployment lifetime with a reduced size and weight of the devices. This is achieved mainly by using the VB-TDMA low-power data upload protocol proposed in this thesis. The mobile sensor nodes, consisting of the GPS and radio modules among others, and the base-stations are powered by batteries, and the optimisation of their energy usage is of primary concern. The presence of the GPS module, in particular its acquisition of accurate time, is used by the VB-TDMA protocol to synchronise the communication between nodes at no additional energy costs, resulting in an energy-efficient data upload protocol for sparse networks of mobile nodes, that can potentially be out of range of base-stations for extended periods of time. The VB-TDMA and an asynchronous data upload protocol were implemented on the custom-designed Prospeckz-5-based wireless sensor nodes. The protocols' performances were simulated in the SpeckSim simulator and validated in real-world deployments of tracking and monitoring thirty-two Retuerta wild horses in the Doñana National Park in Spain, and a herd of domesticated horses in Edinburgh. The chosen test scenario of long-term wildlife tracking and monitoring is representative for the targeted class of applications. The VB-TDMA protocol showed a significantly lower power consumption than other comparable MAC protocols, effectively doubling the battery lifetime. The main contributions of the thesis are the development of the VB-TDMA data upload protocol and its performance evaluation, along with the development of simulation models for performance analysis of wireless sensor networks, validated using data from the two real-world deployments.

Acknowledgements

I would like to offer my sincere thanks to the many people who helped me throughout the duration of my PhD studies.

I must thank my supervisors, DK Arvind and David Robertson, they have been invaluable sources of advice and guidance.

I thank all my colleagues and friends for their help and support. In particular Janek Mann, Anton Shafarenko, Christy Brewster, Rachel Mackay, Charmaine Leitch, Andrew Bates, Hugh Leather, Valentin Radu and Cosmin Dumitrache.

I also thank Rik Sarkar and Robert Stewart for taking the time to examine my thesis and suggest improvements.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Ion Emilian Radoi*)

Table of Contents

Chapter 1

Introduction

Wireless Sensor Networks (WSNs) have made substantial progress recently thanks to the availability of platforms which combine sensors, a low-power processor, and a radio, running wireless protocols for transmitting the data from the sensors to base-stations which bridge to the rest of the internet. WSNs can be used to gather environmental data in a distributed manner at a level of spatial and temporal resolution that was hitherto not possible. Examples of such data include ambient light, temperature, pressure, sound, pollutants in the atmosphere, or dynamic characteristics of mobile entities such as location and movement, which can be used for analysing individual and group behaviour.

Currently, WSNs have a wide range of applications in manufacturing, healthcare, security, military, environmental monitoring (both outdoors and built environments) and mobile-entity tracking [1, 2, 3, 4, 5]. Considering examples of some of these applications gives insight into the vast range of possibilities that sensor networks offer. Examples of military uses include monitoring friendly forces, equipment and ammunition; battlefield surveillance; reconnaissance of opposing forces and terrain; battle damage assessment; nuclear, biological and chemical attack detection and reconnaissance [2, 4, 6]. In healthcare the applications range from patient monitoring and drug administration in hospitals to diagnostics determination. A specific example would be the use of sensor nodes to measure the respiratory rate and activity levels, such as the RESpeck [7] that are used as wireless respiratory patches on the body, enabling doctors to continuously monitor patients remotely. A wide variety of scenarios are considered for built environments, many related to the concept of "Smart Buildings", which consists of allowing people and the environment to be part of a network of devices which combines sensing, processing and wireless networking capabilities. These networks provide detailed monitoring of conditions inside a building in a cost-effective manner. Some examples of scenarios from this class are local adaptation to presence, emergency management, electric device monitoring, CO_2 monitoring, building fabric monitoring, resource tracking, and garden watering [8]. Different applications may be combined in one scenario, such as environmental monitoring and animal tracking applications. These are combined to determine the effect of the environment on the animals, and vice versa. Monitoring the environment has practical applications, such as forest fire detection, bio-complexity mapping of the environment, pollution study, flood detection or agricultural applications [2, 9]. The high-resolution data of the activity and location of animals obtained by tracking and behaviour monitoring applications enables biologists to understand key concepts of animal ethology such as resource use, home range, animal dispersal, and population dynamics [10]. Other applications include full 3D motion capture in real-time using a combination of accelerometers, gyros and magnetometers, and an algorithm which calculates the orientation in 3D space on the devices in a distributed fashion [11, 12, 13]. Traditionally, motion capture uses video cameras, but this extends the range of applications by including scenarios such as motion capture in the outdoors without any infrastructure.

This thesis focuses on one particular class of applications: the long-term tracking and monitoring of mobile entities in the outdoors. This class has witnessed significant growth, as advances in the hardware in terms of size reductions and performance improvements have made many scenarios feasible that were previously impossible. We will examine the challenges faced by this class of applications, propose a solution, analyse the performance of this solution, and compare it to a selection of existing alternatives in the context of a representative scenario.

1.1 The Class of Problems - Long-term tracking and monitoring of mobile entities in the outdoors

In this thesis, we have identified a class of problems, which requires the generation of spatio-temporal data for providing contextual information on the behaviour of mobile entities in the outdoors. This class includes practical, real-world applications, and the main challenges for most of these applications are the following:

- The fact that the applications can involve tracking of living mobile entities such as humans and animals, places strict restrictions on the size and weight of any devices that can be attached to them (henceforth referred to as mobile nodes), which directly limit their battery capacity. Also, deployment in the outdoors suggests designing ruggedised, weather-resistant nodes that can cope with harsh environments. Satisfying these characteristics in the design of the mobile nodes (durable packaging, weather-resistant, waterproof, shockproof / vibration protection) makes them bulkier, which further reduces the space and weight available for the batteries. These limitations need to be addressed by **optimising the battery usage**.

- Tracking in the outdoors normally entails tracking entities in large geographical spaces, usually ranging from tens of kilometres [14, 15, 16, 17, 18] to thousands of kilometres [19, 20, 21, 22]. A practical solution needs to provide good positional accuracy specific to each application, usually ranging from tens of metres to one-two metres, without requiring a huge infrastructure setup (e.g. solutions using RFID tags and detectors, or using video cameras, to cover areas of hundreds of squared kilometres). Reducing the infrastructure requires passing more functionality to the devices attached to the mobile entities. The implication here consists of including **GPS** modules in the mobile nodes.
- A long-term deployment implies continuous tracking and monitoring over periods of several months or years, which requires the solution to have a reliable and robust design, with the ability to recover from any state, and allow access to the data through the course of the deployment. This implies implementing an efficient method for the mobile nodes to **wirelessly upload the data**.

It can be observed that the main challenges of the applications within this class add limitations to the battery capacity of the nodes, while requiring multiple functions such as sensing using GPS, processing and wireless communications, that are demanding in terms of energy consumption. The most important objective of such a system is to extend the battery lifetime, because once deployed, it can be either too expensive or impossible to access the entities to replace the batteries.

Currently there is no solution targeting the entire class of problems, only point solutions focused on specific applications. Even though deployments have addressed certain scenarios in this class, the shortfall of the majority of the these solutions relate to the limited lifetime of the battery-operated mobile nodes. Examples are presented in Section 2.2 which discusses the related work.

The different techniques that have been adopted to tackle specific applications in this class are described in Section 2.2. The approach proposed in this thesis attaches nodes with sensing and wireless communication capabilities to the mobile entities. The sensing capabilities are provided by low-power sensors such as accelerometers and magnetometers for monitoring aspects of activity (for living entities) or handling (for objects), and a GPS module for determining the location. The wireless communication capabilities are used for uploading the collected data to an infrastructure of base-stations (data collection nodes), which are placed at strategic locations so as to maximize their contact with the mobile nodes.

Figure 1.1 is illustrating the concept of a deployment of wireless sensor nodes attached to mobile entities and an infrastructure of base-stations for collecting the data. The mobile nodes only upload data when in range of a base-station. The number of deployed base-stations should be chosen taking into account the size of the area that requires covering and the coverage level required by the application (better coverage can lead to shorter network response times).

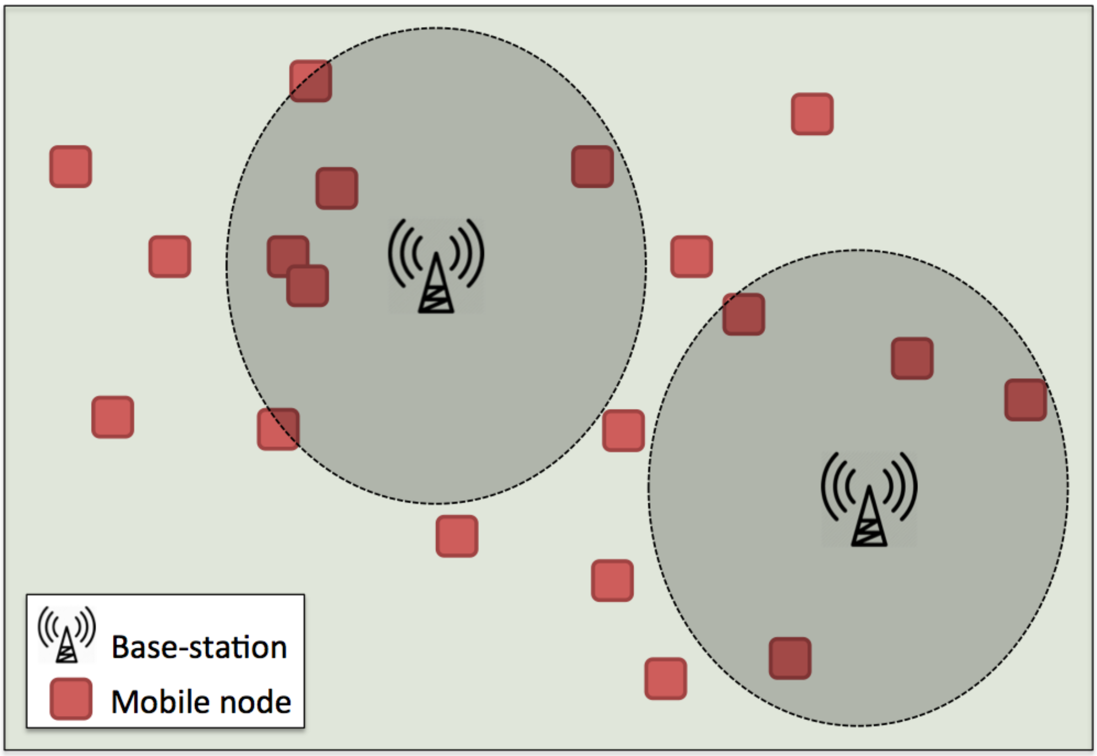


Figure 1.1: An example of mobile nodes and two base-stations with their radio coverage highlighted.

The novelty of our approach is in the VB-TDMA data upload protocol (for details please refer to Section 3.1.2), which uses the time provided by the GPS, obtained with no extra energy cost, to synchronise a network of battery-operated, radio-enabled devices. The protocol uses this synchronisation to implement a TDMA scheme designed to satisfy the node communication needs of the applications within the target class. This TDMA scheme is only used for the exchange of packets containing sensor data. This enables the protocol to significantly improve the battery lifetime of the devices by eliminating the major energy wastage sources of wireless data upload processes: collisions, overhearing, control packet overhead and idle listening. Since the VB-TDMA eliminates the synchronisation overhead completely, then at least theoretically, no existing MAC protocol can be as, or more energy efficient, as they would have some overhead for synchronising the communication. Also, the synchronisation overhead would be high in the case of most applications in this class, as they form sparse networks of mobile nodes with low data rates, and the mobile nodes are out of reach of the base-stations for extended periods of time. This thesis demonstrates that the VB-TDMA is significantly more power efficient than other low-power MAC protocols and can potentially double the deployment lifetime. The performance of the proposed approach was demonstrated and validated both in real deployments and in simulations, using a representative scenario for this class of applications.

The class of tracking and monitoring applications considered in this thesis includes a growing number of different scenarios. For example, consider the millions of shipping containers moving around the world. The life cycle of a container's journey begins when it is loaded with goods and sealed until arrival at the destination warehouse. The container is transported to a port, loaded onto a container ship for the destination port, and unloaded onto a truck trailer or a train for the journey to the final destination. Throughout this journey the containers are tracked and the sensors monitor the handling during the transportation, loading and unloading processes. This data can be uploaded wirelessly to base-stations located on the container ships, ports and warehouses which in turn upload the data to a centralised database.

Another such example is a self-service, bicycle-sharing scheme for short journeys in major cities, using bicycles equipped with sensors to measure air quality. The sensors measure pollutants such as particulate counts PM_{10}, $PM_{2.5}$ and $PM_{1.0}$ values, and noxious gases such as NO_2, O_3, and CO, together with temperature and relative humidity, and all the values are tagged with spatiotemporal information. The bicycles are used to crowd-source sensor data during their journey and the data is uploaded to a central server via base-stations located at the bicycle-stations or at waypoints.

Another example is the long-term tracking and monitoring of horses in the wild, where the deployed system has to operate without any external interference during its lifetime. This scenario was chosen as the main test scenario in this thesis, and its challenging requirements are described in Section 2.1.1. The principal goal is to maintain a challenging GPS sampling rate of once every approximately twenty minutes over a period long enough to include seasonal variations and to upload this data wirelessly, while respecting the restrictions on the size and weight of the mobile devices. The best off-the-shelf solution for animal tracking with GPS collars is provided by Vectronic Aerospace [23] and was launched in 2014. The company offers two products, the Vertex Survey Globalstar [24] and the Vertex Survey Iridium [25]. The Vertex Survey Globalstar collar, according to its datasheet, has a lifetime of 2.9 years when sampling the GPS twice per day, or 3.7 years when sampling once per day. This performance is inadequate for a scenario such as the tracking and monitoring of the wild horses described in Section 2.1.1 (which requires 72 GPS locations per day), and the weight of the sensor node without the strap is 340g (double the weight of the nodes proposed in this thesis). The second product, the Vertex Survey Iridium, comes with three different sizes of batteries for the mobile node, weighing 260g, 385g and 520g without the strap. The datasheet expresses the nodes' lifetime as the average number of GPS positions collected, but the stated performance is achieved in a controlled environment (in terms of ambient temperature and other parameters) which is unlikely to be encountered in real outdoor deployments. For the three battery sizes, the node can collect 4500, 11300 and 23900 GPS locations over its lifetime. With our solution, using the VB-TDMA data upload protocol, we have recorded a comparable performance (23890 GPS locations) to the 520g Iridium collar, while reducing the node weight to 165g, which is more than **three times lighter**.

1.2 Contributions

The thesis proposes a systemic approach to providing a solution for long-term tracking of mobile entities in the outdoors, which considers the technical choices in the hardware design and communication protocols for uploading the data. The solution was demonstrated and tested both in simulations as well as in real-world deployments. The technological challenges for such deployments are driven by the high performance expected of the devices for acquiring high-resolution data, the size restrictions, and the deployment lifetime expectancies. These challenges have to be addressed at both levels: the hardware design of the nodes, and the design of the protocols for transferring efficiently the sensor data from the mobile nodes to the base-stations. As the custom-designed Prospeckz-5 hardware platform (presented in Section 2.1.3) benefits from state-of-the-art low-power components, the main challenge addressed in the thesis was to design energy-efficient communication protocols for uploading the data wirelessly from the mobile devices to the base-stations.

The key contributions of this thesis are:

- The development of the VB-TDMA, an energy efficient data upload protocol designed specifically for the targeted class of applications. The protocol was designed to be easily tuned to satisfy the specific requirements of applications within the target class, by configuring its different parameters (details in Section 6.2). In addition to prioritising the minimisation of the energy consumption, its design also took in consideration features such as flexibility and scalability in order to accommodate network changes, including network growth (see Section 3.1.2.1). The experiments presented in this thesis show the superior performance, in terms of energy efficiency, of the VB-TDMA protocol compared to other low-power MAC protocols, leading to twice as long deployment lifetimes. The low-power nature of the protocol combined with the Prospeckz-5 low-power hardware offers similar results in terms of the number of GPS locations sensed per node lifetime to the state-of-the art animal tracking collars [25], but with three times lighter mobile nodes.
- The development of simulation models for performance analysis of wireless sensor networks, including battery lifetimes. The hardware models reflect the performance and behaviour of the hardware components of the Prospeckz-5-based nodes (MCU, radio, accelerometer, magnetometer, GPS, battery). The data-mirror mobility model uses real movement data to offer more realistic network performance predictions compared to trace-based mobility models, in certain scenarios (e.g. the movement of the wild horses).
- The validation of protocols and simulation models in two real-world deployments of a representative scenario for the target class of applications, along with the presentation of the experiences and lessons learned from these deployments:
  - One deployment took place in a nature reserve in Spain, marking thirty-two wild horses, and having eight base-stations.
  - The other, in the paddock of the School of Veterinary Studies of the University of Edinburgh, marking eight domesticated horses and having one base-station.
- The performance analysis of the VB-TDMA protocol, including sensitivity tests

for parameter selection, scalability tests, evaluating the impact of enabling the multihop / store-and-forward functionality, and comparisons for twelve-month simulated deployments of the VB-TDMA against other protocols deemed most suitable for this class of applications.

- Insights into a real-world problem on the group behaviour of wild horses using mobile wireless sensors.

1.3 Publications

The work presented in the thesis has been the basis for the following publications by the author:

I. E. Radoi, J. Mann, and D. K. Arvind "The VB-TDMA for Long-term Tracking and Monitoring of Mobile Entities in the Outdoors". To be submitted to ACM Transactions on Sensor Networks (TOSN).
The paper presents the validation of the simulation models used in the performance analysis of the VB-TDMA, along with the protocol's scalability study and comparison to other candidate low-power MACs with a store-and-forward capability added.

I. E. Radoi, J. Mann, and D. K. Arvind "Performance Evaluation of the VB-TDMA Protocol for Long-term Tracking and Monitoring of Mobile Entities in the Outdoors". In Proceedings of the 11th ACM International Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet), ACM, 2015.
The paper [26] presents the performance evaluation of the VB-TDMA protocol in simulations, along with comparisons to other low-power MAC protocols deemed suitable for the targeted class of applications.

I. E. Radoi, J. Mann, and D. K. Arvind "Tracking and Monitoring Horses in the Wild Using Wireless Sensor Networks". In Proceedings of the 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2015.
The paper [27] introduces the VB-TDMA protocol, and presents two real-world deployments. It describes the experiences from these deployments and presents results on the network performance and animal behaviour.

J. Mann, I. E. Radoi, and D. K. Arvind "Prospeckz-5 - A Wireless Sensor Platform for Tracking and Monitoring of Wild Horses". In Proceedings of the 17th EUROMICRO Conference on Digital System Design (DSD), IEEE, 2014.
The paper [28] presents the custom-designed Prospeckz-5 sensor platform for tracking and monitoring wild horses, and the designs of the mobile node and the base-station, which together form the wireless data upload process.

D. A. Crisan, I. E. Radoi and D. K. Arvind "CoAP-Mediated Hybrid Simulation and Visualisation Environment for Specknets". In Proceedings of the ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (PADS), ACM, 2013.
The paper [29] presents the SpeckSim behavioural simulator, which was extended to support CoAP-enabled hybrid simulations and 3D visualisations. SpeckSim is the simulator of choice for this thesis, and the paper describes its architecture and models: hardware and device models, battery model, microcontroller and clock models, radio model, protocol models, application behavioural models, communication models, movement models and environments.

1.4 Previous Work

The Prospeckz-5 hardware platform presented in this thesis was designed by Janek Mann, and the TinyOS port for the EFM32 Gecko microcontroller (used on a different hardware platform aimed at motion sensing applications), along with an initial TinyOS implementation of time synchronisation based on GPS time pulses was provided by Anton Shafarenko, both researchers at the Centre of Speckled Computing, the University of Edinburgh. The implementation of the asynchronous protocol presented in this thesis on the Prospeckz-5 was performed by Janek Mann. The SpeckSim simulator developed by Ryan McNally was extended and enhanced with new models in this thesis.

1.5 Structure

A summary of the rest of the six chapters in this thesis is presented below.

Chapter 2 presents the test scenarios, the custom-designed hardware platform at the core of all nodes, and the SpeckSim behavioural simulator for wireless sensor networks. It also presents the current state of the art in terms of tracking mobile entities, highlighting the existing challenges for this class of applications.

Chapter 3 introduces the proposed data upload protocol for the targeted class of applications, the VB-TDMA, along with the evaluation environment developed for validating and characterising the complete solution, consisting of the implementation of the hardware platform simulation models and the mobility model in the SpeckSim simulator.

Chapter 4 presents the application of our proposed solution for a representative scenario for the target class of applications, the tracking and monitoring wild horses. It describes the node designs and the pre-deployment tests, presents a long-term deployment on thirty-two wild horses in a nature reserve in Spain that uses an asynchronous protocol for the data collection process, and a smaller deployment on eight domesticated horses belonging to a university's teaching herd that uses the VB-TDMA protocol. For both deployments, an analysis of the sensor data is provided, giving insights into the individual and group behaviour of the horses.

Chapter 5 validates the simulation models and the data upload protocols' implementation in the simulator against data from the deployments, and uses these models to predict the network performance. It presents a performance evaluation of the VB-TDMA protocol, including comparisons to a selection of existing low-power MACs in the context of the targeted class of applications.

Chapter 6 summarises the presented work and discusses the contributions of the thesis. It offers suggestions for future work and improvements for enhancing the performance of networks running the VB-TDMA.

Chapter 2

Background and Related Work

Tagging sensor data with spatio-temporal information enables a wide range of applications. Challenges arise when it is required to do this over extended periods of time using wireless-enabled battery-powered devices constrained in size and weight, and without having the option to change batteries during the system's lifetime. Existing solutions for specific scenarios from the chosen class of applications, long-term tracking and monitoring of mobile entities in the outdoors, face limitations due to the two main energy consuming processes: the spatio-temporal tagging of the sensor data and its wireless upload. This chapter introduces the chosen test scenarios, the custom-designed hardware platform at the core of all nodes, and the SpeckSim behavioural simulator for estimating the performance of wireless sensor networks. It also places the work in this thesis in the context of the current state of the art technology in terms of tracking mobile entities (with a focus on animal tracking and monitoring) and the existing complementary MAC protocols, hardware simulation models and mobility models.

2.1 Background

This section introduces two representative scenarios for the class of applications of long-term tracking and monitoring of mobile entities in the outdoors, the hardware platform used in the mobile and base-station nodes built for tracking wild horses, and the SpeckSim simulator for wireless sensor networks.

2.1.1 Main Test Scenario - Tracking and Monitoring Wild Horses

The main real-world application considered in this thesis is the long-term deployment of sensor platforms for tracking and monitoring horses in the wild. In this case, long-term represents a period longer than six months, ideally twelve months, aiming to include the effects of seasonal variations. The requirements include collecting GPS positions and light intensity approximately every twenty minutes along with the activity and head orientation over that interval, and uploading this information wirelessly to a network of static and mobile base-stations. These requirements are demanding, considering the size and weight restrictions (few hundred grams including the batteries) for mobile sensor nodes.

The Retuerta wild horses that were monitored are one of the oldest horse breeds in Europe. The breed does not relate to any of the other breeds studied so far and is genetically distant from them. It may be the ancestor of the remaining horse breeds in the Iberian Peninsula. Its name refers to the habitat where the horses live in summer. The retuerta is the contact area between the marshlands and the dunes in Doñana National Park, which does not dry in the summer. From an original stock of only five individual rescued by Doñana Biological Station thirty years ago, 150 individuals (with a minimum of 3/4 purity) survive today within the limits of Doñana National Park. Therefore, monitoring these horses to understand their behaviour is important for the future management of the herd.

For this particular deployment, the locations for the static base-stations were chosen by the nature reserve custodians to be close to the paths normally taken by the horses. Fortunately, Doñana benefits from towers scattered in the reserve, which represent ideal locations for the base-stations as they provide power, ethernet connections to a local private network, and an elevated position for mounting the antennas. In order to avoid packet collisions and to provide redundancy, adjacent base-stations operate on different channels (one of four), as illustrated in the left-hand side of Figure 2.1.

2.1.2 Cyclists Gathering Air Quality Data

In this scenario air quality data is sourced from cyclists riding bicycles equipped with air quality sensors. As an example, we considered students equipping their bicycles with sensors for collecting air quality data (particulate levels, NO_2, O_3, temperature and relative humidity). The data collected is uploaded wirelessly to base-stations which are distributed in locations around the university campus. This scenario was

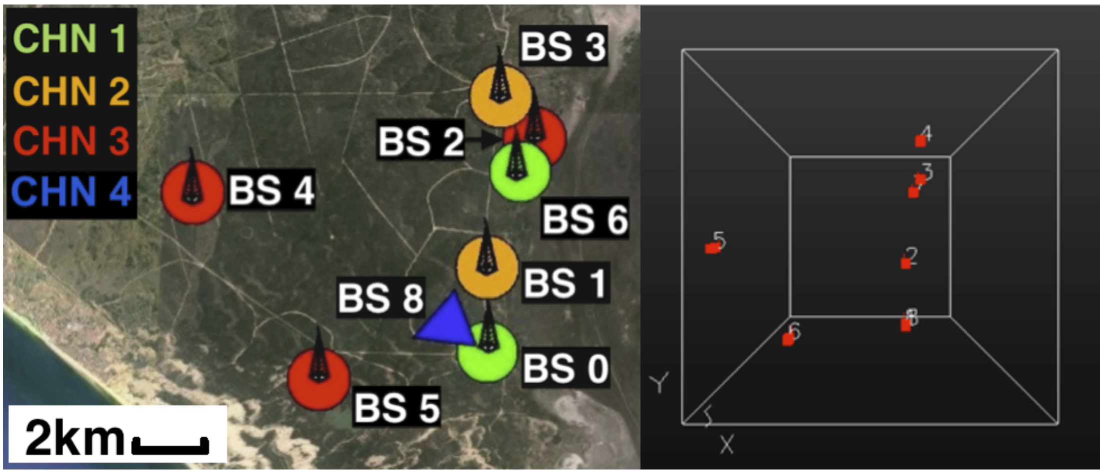


Figure 2.1: Base-stations deployed in Doñana (left); base-stations in SpeckSim (right).

simulated in the SpeckSim simulator (as shown in Figure 2.2), with ten mobile nodes representing the student bicycles, and three base-stations for data collection, located at the halls of residence. The mobile nodes follow a circuit starting at the halls of residence and visiting waypoints representing lecture theatres, a coffee shop, and the university gym, over a typical 24-hour period. The bicycles exchange data at any of these locations or in the streets when within range. Since the pollution levels are time-sensitive, the freshness of the data is important for applications that aim to display air quality information close to real time. An example would be an application that chooses a route to a given destination based on the best air quality. It directly benefits from having as fresh air quality data as possible by crowd sourcing this information from cyclists.

2.1.3 Prospeckz-5 hardware platform

The Prospeckz-5 hardware platform [28] (Figure 2.3) was designed to fit in the lid of a hard and ruggedised off-the-self enclosure [30]. The board has the following hardware characteristics:

- 128KB RAM, 1MB ROM
- EFM32 Microcontroller, Cortex-M3 (Run Mode 200μA, Deep Sleep 1.1μA)
- NRF24L01+ radio, operating at 2.4GHz, chosen for its high data rate of 2Mbps allowing for short on-air times, minimizing energy consumption per bit (Tx 11.3mA, Rx 13.5mA @ 2Mbps air data rate)

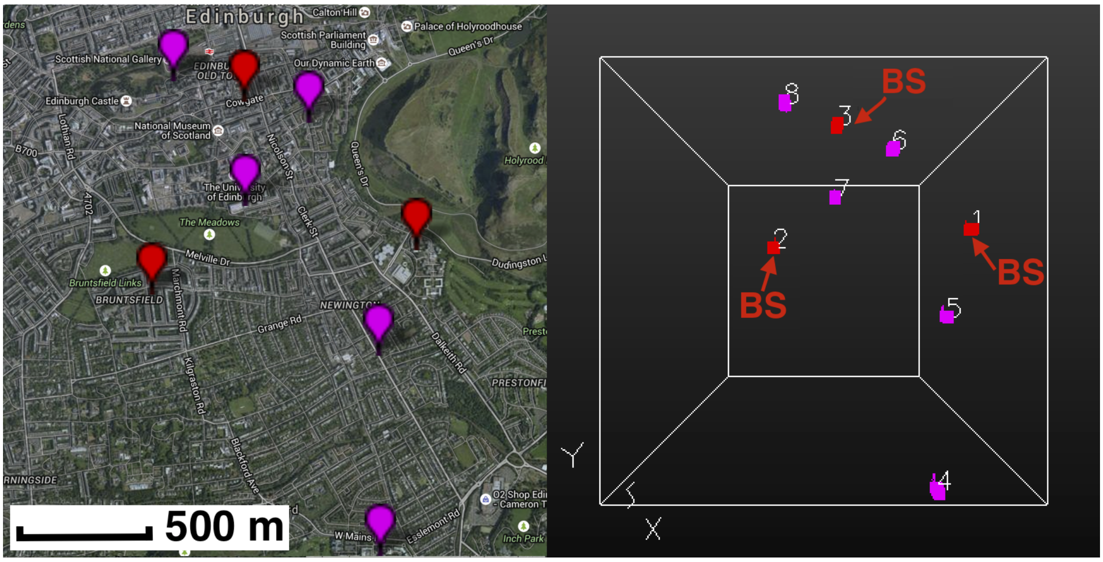


Figure 2.2: Base-stations and visited locations on the map (left), and in SpeckSim (right).

- RFX2401C radio amplifier, providing 24dBi end-to-end gain (Tx 17mA, Rx 10mA)
- External interfaces: I2C, SPI, GPIO, Analog
- 8MB flash chip
- Solar battery charging circuit
- Dual battery interface
- FTDI compatible header for RS232
- Sensors
  - GPS - Fastrax UC430 (Active 47mA, Tracking 37mA, Hybernate 20μA)
  - Accelerometer - Freescale MMA8451Q (6μA - 165μA)
  - Magnetometer - Freescale MAG3110 (ODR 10Hz 137.5μA)
  - Light - Maxim Integrated MAX44007 (0.65μA)

The board offers the possibility of attaching external sensors to it, and, in order to obtain better radio range for the communication between nodes, it was designed to have a radio amplifier and an external antenna connector. A block diagram of the platform is presented in Figure 2.4.

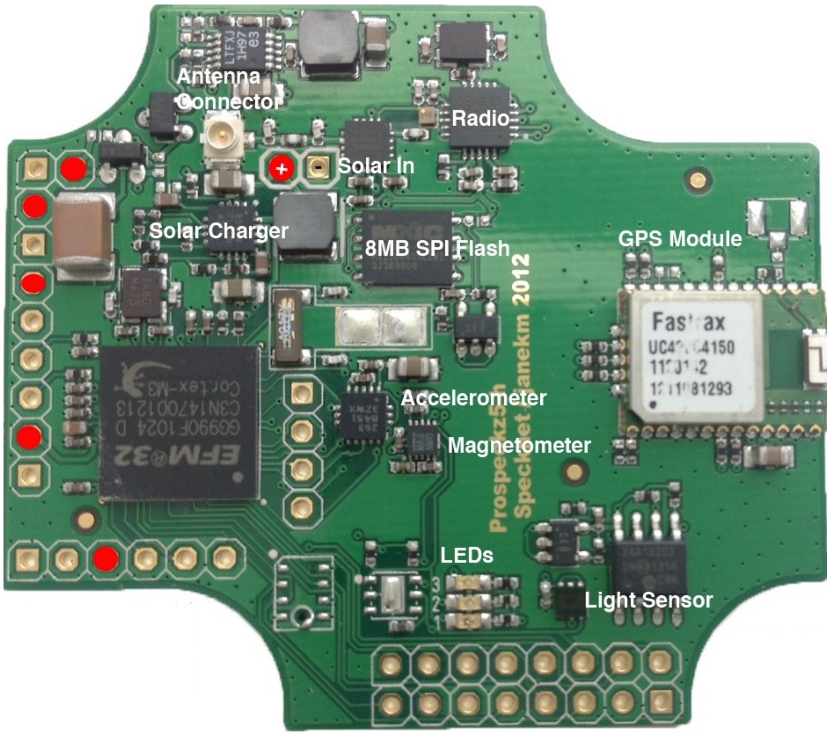


Figure 2.3: The Prospeckz-5 hardware platform (54x48 mm).

2.1.4 The SpeckSim Simulator

The chosen simulation environment is SpeckSim [29], a discrete-event behavioural simulator for modelling computation in networks of sensor nodes. It was developed for algorithmic-level simulations of network behaviour, for different hardware and protocol configurations of static and mobile sensor devices with computational and networking capabilities.

The hardware platform used in the actual deployment on wild horses was modelled in SpeckSim and validated using real data (see Section 5.1.1).

SpeckSim is written in Java and includes models of devices, networks and deployment environments described at different levels of abstraction, ranging from algorithms to faithful representation of standards-based protocols. The simulator was originally developed as a behavioural level network simulator for use as a tool in development and analysis of distributed algorithms for localisation and routing in wireless sensor networks [31, 32]. Further development has incorporated lower layer communication protocols, radio channel models and modelling of hardware for analysis of power consumption and resource usage.

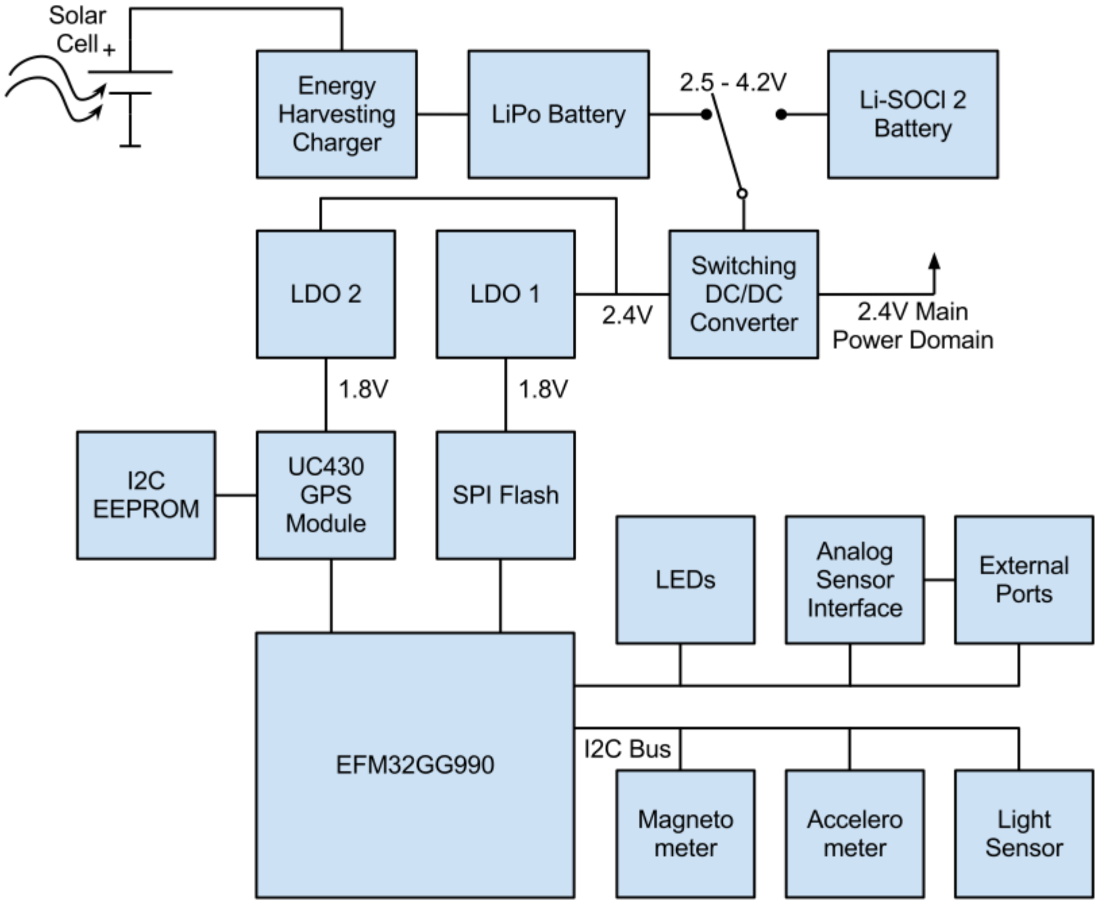


Figure 2.4: Prospeckz-5 block diagram.

2.1.4.1 Architecture

The core of the SpeckSim simulator is the event queue. Any model-related computations are initiated by an event and the global simulation time progresses as events are popped from the queue. All the activities of the simulated devices are therefore asynchronous.

The architecture of the SpeckSim simulator, as shown in Figure 2.5, has four main components:

- **Simulation Core:** The Simulation Core contains the core functionality of the simulator and the three top-level categories of simulation models: Devices (Specks), Environments and Movement models.

- **Simulation State:** The Simulation State is a snapshot of the state of the Simulation Core at specific instants, containing the current state of the models being used in the simulation. The state of the simulation is extracted to produce visualisations and statistics of an executing simulation within the GUI.

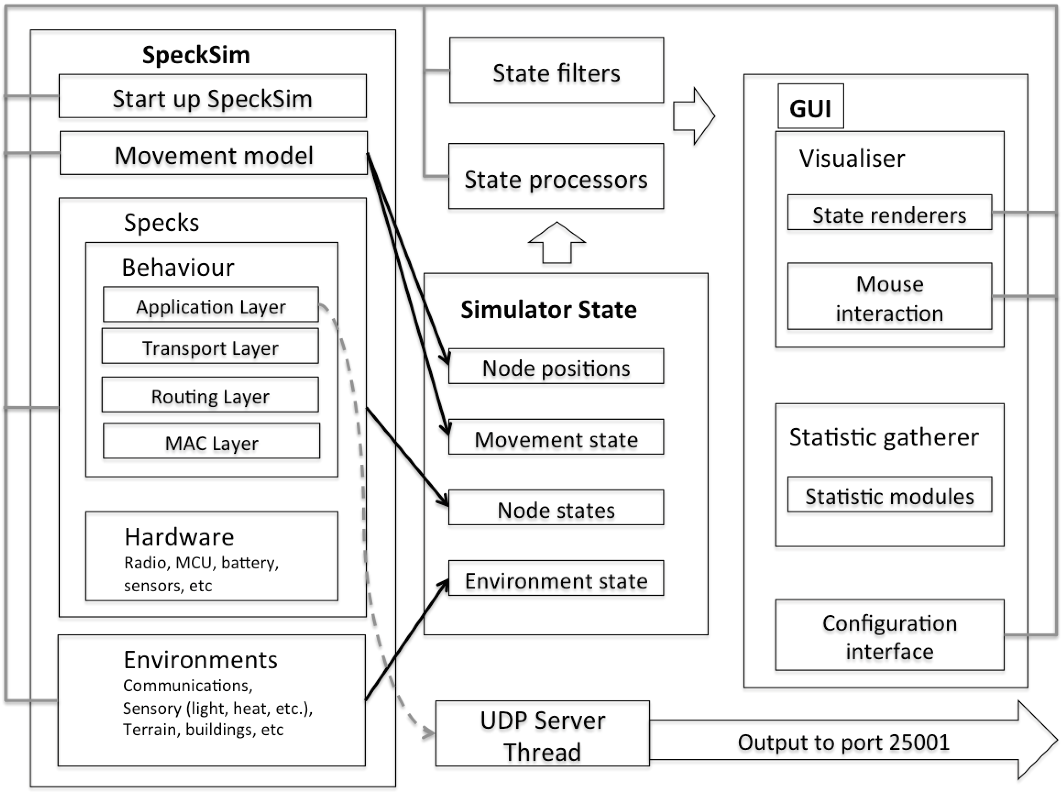


Figure 2.5: SpeckSim architecture.

- **Statistics and Data Processing:** The simulator includes a number of statistical and data analysis modules to extract the results of interest from a simulation. These modules execute analysis algorithms on the state of the simulation either as the simulation is running or after completion. The simulator provides analysis of device power consumption and battery lifetimes, MAC and routing protocol performances (delivery ratio, latency and throughput), communication channel performances (collisions, packet loss and contention levels). Higher level and application specific behaviours can also be analysed by extending existing classes.

- **Graphical User Interface (GUI):** The SpeckSim user interface is automatically generated from annotations made to the Java simulation models and allows users to configure any variable parameter of a simulation model as specified in the implementation. An OpenGL 3D visualisation of the simulation is also provided. Renderer classes provide visualisations; existing renderers include those displaying devices, terrain, communication links and device's positional estimations. Additional visualisations may be added to display interesting aspects of the simulation state such as 3D visualisation of the scenario, log viewers and configuration panes showing menus for configuration of the simulation models.

2.1.4.2 Simulation Models

Hardware and Device Modelling. Devices are the top layer of simulated entities and essentially model wireless sensor devices. The simulator may be extended to model the behaviour of other types of devices, such as application servers, mechanical and actuation devices. Each device object contains an instance of models of the hardware, software and communication protocols required by the simulated physical device. There is no operating system for devices within the simulator; high level application specific operation is termed as Behaviour and may be implemented directly within the class describing a device type or within Behaviour classes. The principle function of hardware component models is to enable simulation of the energy usage of battery-operated sensor devices.

Battery Model. Battery models simulate the effect of drawing current from a battery over periods of time. The simulator includes models for several types of batteries and a cell that can be configured for any desired capacity and voltage.

Micro-controller and Clock Models. The simulator includes models for a micro-controller and clock system. The MCU model provides current consumption figures for typical MCU states such as sleep, power down and active and is used in current consumption analysis. The clock model simulates the real-time clock aspects of a typical micro-controller and allows for the effects of clock frequency drift to be simulated. The clock also provides a real-time scheduling interface for simulated software components to use in scheduling events.

Radio Model. Radio models provide the devices with access to wireless communication channels and also simulate the radio device hardware. There are several models in the simulator:

- CC2420 - A model of the Texas Instruments CC2420 2.4 GHz radio transceiver
- CC1000 - A model of the Texas Instruments CC2420 sub-1 GHz radio transceiver
- Selectable Transmission Shell Radio - A simple radio model that enables simulation of range- and shape-controlled transmissions. Using this model, users can specify the range and 3D shape of a transmission made by the radio (devices lying within the shell may receive transmissions). This can be configured with

many different shells modelling different antenna designs and orientations. The simulator is extensible and additional radio models can be included in the future.

- A model for the Nordic NRF24L01+ was added to this selection. The radio's model along with the other hardware simulation models resembling the Prospeckz-5 components are presented in Section 3.2.1.

Protocol Models. Devices may use a communication stack made up of different protocol models. Protocol models are stacked to pass transmissions and receptions from application level behaviour to a radio via routing and MAC protocols and vice versa.

Application Behavioural Models. A device may have many top-level behaviours allowing for simulation of multi-purpose devices. A device's behaviour has access to all the components of the device such as protocols and radios for communications, the device clock for real-time scheduling and the battery model for power-aware behaviour.

Communication Models. Communication Models simulate the properties of wireless and wired communication channels. The model used in a simulation is dependant on the hardware models configured in the simulated devices. For example, the transmissions of a wireless device configured with a transmission shell-based radio will be passed via the simulation mode for calculating interference and reception of such radio models whereas the communications of a device configured with a model of a real radio such as a CC2420 or CC1000 will be handled via the communications model of free space path loss and transmission power based contention.

Movement Models. Movement models specify the position and orientation of devices within the simulation environment. The simulation environment uses a floating point, three-dimensional position system. Movement models are required to provide 3D positions, and if required velocity vectors for each simulated device. Multiple movement models may be used within a single simulation to allow different devices to use different models. However, all movement models share some common configuration functionalities such as: 2D/3D modes, fixed and randomised orientations, maximum device velocity control and interaction with Environment models such as those modelling ground terrain and buildings to position devices according to the topology of the ground or a building within the simulation environment.

Environments. Environment models are used to simulate any global parameters within a simulation; the variety of models and parameters are diverse and include:

- World: provides configuration of the simulation world such as the scaling factor applied to the simulators floating-point positional reference frame.
- Failure: controls the failure rate of devices within the simulation, allowing for specific, randomised or position based failure of individuals of groups of devices.
- Radio Impediments: allows impediments and interference sources to radio transmissions to be positioned within the simulation environment.
- Terrain: a model of the terrain of the environment that can take height map data from image or standard csv formats.
- Building: models buildings that are specified by inputting architectural drawings showing wall positions. The walls are added as radio impediments and can be used in a specific communication model for indoor environments.
- Environmental conditions: Conditions such as Water Level, Humidity, Temperature and Gravity are modelled in individual environments. These models can be configured to simulate the effects of changes to such conditions.

2.2 Related work

Tracking and monitoring applications using sensor devices have been studied for several decades. The recent tracking methods differ significantly between outdoor and indoor applications [33, 34], due to the availability of GPS technology which is now widely used for outdoor tracking.

Indoor tracking applications use techniques such as Pedestrian Dead Reckoning (PDR), Activity Recognition, Particle Filter and WiFi Fingerprinting. There are also hybrid localisation solutions that combine PDR with WiFi fingerprinting, such as UnLoc and HiMLoc [35, 36, 37]. The inertial sensors (accelerometer and gyroscope) used by the indoor tracking solutions can also be used for activity recognition [38].

Recent outdoor tracking applications use GPS for getting good positional accuracy (within few metres), but before GPS receivers became both low-cost and available for mote-scale devices, other tracking methods were used. In this regard, Amundson and Koutsoukos [39], present tracking methods for WSNs that are based on sending and processing different signals. These methods can use: radio signal propagation, ultrasound or audible wave propagation, infrared signal attenuation, radio frequency propagation, or light beacons. Radio Frequency Identification (RFID) transmitters and fixed detection nodes was one of the most commonly used tracking method. An example of a successful deployment using this method is presented by Dyo et al. [40] consisting in tagging badgers with RFID transmitters in order to analyse their social co-location patterns over a period of one year.

Non-GPS solutions were used for many tracking scenarios, such as tracking busses [41], packages, or perishable goods (e.g. meat) [42, 43, 44]. Even though GPS is the most commonly used technology in new bus tracking installations, other methods exist, such as Signpost and Odometer (SO) and Radio Navigation. SO was at one time the dominant vehicle location technology. It uses radio beacons mounted on top of utility poles, and when a bus passes a signpost, its position is relayed back to the control center. The system has several drawbacks, such as being costly to install and requiring a high degree of maintenance. Radio Navigation was used even as a commercial service for locating vehicles (e.g. Teletrac in Los Angeles, USA). It uses low frequency waves to provide coverage, and it determines the location based on the reception of transmissions and the associated timing. Several methods for tracking and monitoring perishable goods were considered, such as a time-temperature indicator system for tracking the thermal exposure history of temperature sensitive perishable products [43], tracking the production history of food products to enable the verification of product origination and to trace back the source of pathogens [44], and a RFID based data capture system that can help solve the problems associated with the logistics of short shelf life products [42].

Since the class of applications considered in this thesis requires tracking and monitoring in the outdoors, and due to the availability of compact, low-power and low-cost GPS receivers, the focus throughout the thesis is on solutions that use GPS for locating mobile entities.

The rest of this section presents the current state of the literature for animal tracking deployments using sensor devices, MAC protocols for WSNs, WSNs hardware simulation models and mobility models.

2.2.1 Animal / Wildlife tracking

A variety of deployments for tracking and monitoring different types of animals on the ground [45, 19, 16, 17, 18, 15, 20, 21, 22, 46, 47, 48], in the air [21], underground [40, 14, 49] and underwater [50], have been relevant in informing strategy for the chosen test scenario in this thesis. The research in animal tracking and behaviour monitoring applications falls into two broad categories: non-GPS animal tracking and animal tracking based on GPS. As the main test scenario in this thesis is the long-term tracking and monitoring of an endangered species of wild horses, our focus on the related work lies more on the solutions for terrestrial deployments for monitoring animals, especially the tracking solutions that use GPS for determining the locations.

2.2.1.1 Non-GPS Animal Tracking

Chen Liu et al. [22] present a solution for tracking monkeys in the wildlife reserve of Qiling Mountain, China. Their locations are determined by analysing the sounds that the monkeys make along with using video surveillance, and further applying pattern recognition on the audio-visual data. Markham et al. [14] suggest using magneto-inductive localisation for underground tracking of burrowing animals (badgers). Collars are attached to the badgers, containing nodes with sensors that measure magnetic field strengths. The recorded data is stored locally until the animal surfaces, at which point it is uploaded using a 2.4GHz radio. Using this method, besides localising the animals, the structure of the underground tunnels can also be determined and mapped. Osechas et al. [49] describe a WSN deployment for tracking the movement of rats attached with Mica2dot nodes, along with a sensor suite of a microphone and an accelerometer. Rats usually live in underground burrows, an environment where radio communication is extremely limited, thus having base-station nodes placed on the surface, at the entrances to the burrows. The network is designed to run a multi-hop routing algorithm on top of a MAC protocol, which enables data exchange when rats meet in the burrows. Thus, when a mobile node comes in contact with a base-station, it can upload its data along with the data gathered from other mobile nodes. Johnson and Tyack [50] present DTAG, a device designed to monitor the behaviour of whales in the context of their dive cycle. This consists of sensors for measuring in a synchronised manner the sound and the orientation of the whales. Araki et al. (United States patent [45]), present a solution for animal-herd monitoring using WSNs. Their method only tracks the location of the entire herd, and not the individual members, and is more suitable for use in domestic herds. Dyo et al. [40] present a study of a wildlife monitoring network deployment meant to analyse the social co-location patterns of badgers over a period of one year. This was achieved by using collars with RFID transmitters and fixed detection nodes placed at key locations throughout the woods. The deployment was expanded to also use fixed sensor nodes to monitor environmental conditions such as temperature and humidity, in order to determine their effect on species migration and movement patterns. Also, a gateway with 3G connectivity using solar power was placed in a convenient location so that it can be easily accessed and maintained. The network collects three types of data: RFID reading for badger tracking, environmental data (temperature and humidity) at regular intervals, and network health data (battery levels, memory usage and sensor errors). While the data collected by the detection nodes and the environmental monitoring nodes is compressed and stored locally, the network health data is routed towards the 3G gateway, along with other low-volume data that is considered to be critical. Even though the described deployment lasted for one year, the proposed solution's main limitation, as in the case of any RFID-based solution, is the confinement to a small/narrow surveillance area that needs to be filled with static RFID receivers. If the surveillance area would be wide and the RFID transmitters would not get in range of the RFID receivers, then no tracking information would be collected.

Several research projects have attempted the development of similar technologies and methods for wildlife tracking [51, 52, 53, 54, 14, 55, 56] including reverse-GPS technologies [57, 58] which estimate the animals' locations from the time of arrival (TOA) of a radio signal emitted by the tags worn by these animals. Reverse-GPS methods have the advantage of being low-cost and having less intrusive tags (making the surveillance of smaller animals possible), but are still limited in terms of the size of the geographic area that can be covered (e.g. tens of kilometres). Similar to these methods are systems based on the automated angle of arrival (AOA) [53, 59]. However, these also have disadvantages: the accuracy degrades linearly with the distance from the receivers; and the tags are more costly due to multiple antennas and the afferent circuitry for handling the signals. Weiser et al. [57] characterize and validate the accuracy of reverse-GPS wildlife tracking systems, and propose a new system called ATLAS. This system is an improvement on a similar previous system [58] and, also on automated AOA systems, due to the addition of two main features: the extensive use of beacon transmitters at known locations, and providing a method for quantifying the uncertainty in each arrival-time measurement from the signal-to-noise ration.

2.2.1.2 Animal Tracking Solutions Based on GPS

For the past three decades, ecological studies have been using terrestrial radio tracking methods [60, 16]. This has changed with the advent of GPS technology which revolutionized wildlife tracking [10, 61]. GPS receivers are able to provide accurate location coordinates (with errors of few metres) at low power consumption and their size can be scaled down to a few millimetres (e.g. Fastrax IT430 GPS receiver: 9.6 x 9.6 x 1.85mm with a 500μA power consumption [62]). Therefore, GPS technology for wildlife tracking became a feasible option for obtaining accurate position coordinates over an extended period of time, opening new perspectives for wildlife study.

The research presented in [63] is aimed at testing the behaviour and performance of GPS receivers for wildlife monitoring applications. A considerable amount of previous research has concentrated on measuring the performance of GPS collars for animal tracking in terms of location accuracy, by placing them on a free-ranging moose [16, 17] and on five white-tailed deer [18]. In a 2003 publication [19], Anderson and Lindzey describe a six-to-eight months deployment of collars containing GPS receivers, which were attached to eleven cougars. The deployment was performed in a 2,120km^2 forest in the USA and had the purpose of studying predation rates and identifying the prey-selection patterns. Each collar was programmed to get six GPS positions per day (mostly during evening and night time). There was no network infrastructure, all the data was stored locally on the nodes. The data was collected from the nodes upon death of the animals or recapture. The study reported no sign of injuries caused by the collars to the animals. By having no radio communication, and by acquiring a low number of GPS position fixes, the deployment lifetime was six-to-eight months. This method of only storing the data locally on the nodes does not offer the possibility of accessing any of it without recapturing the animals. Since the capturing of wild animals cannot be done excessively, this leads to accessing the data only at the end of the deployment. Thus, this method is not appropriate for applications that require close to real-time access to the data.

The ZebraNet system [20] consisted of a thirty-node WSN deployment for localising zebras in their natural habitat across an area of 400^2 kilometres for one year. This was one of the early examples of using sensor nodes to track animals. The nodes were attached to the animals using collars, and established peer-to-peer connections to route information to mobile collection points (base-stations). This way, researchers can drive by and collect data from many animals, even though they only encounter few. The design of the application consisted in sampling and logging GPS positions every three minutes, and logging temperature, weather information, environmental data and body movements, every hour. In order to maintain this data collection rate over one year, each node weighed 1151 grams, the bulk of which was the battery and the solar array charger. The node design heavily relied on the solar cell array, as the battery would only last for a maximum of five days without being recharged. Besides a GPS module, 1MB of Flash and a CPU, the node was equipped with two radios: one for short-range (up to 100m) and another for long-range (up to 8km) communication. In contrast, the solution proposed in this thesis is focused on energy efficiency, having significantly smaller and lighter nodes (165 grams, 7 times lighter), which can be used in long-term deployments while powered only by primary cells without requiring recharging. This is achieved both by using newer low-power hardware components (due to the advancement of the sensor technology in the eleven-year gap between the two hardware designs) and lower-power data upload protocols. The paper also presented a protocol evaluation in simulation, based on storage constraints, bandwidth constraints and energy trade-offs. Two protocols were proposed. A flooding protocol, used to move the data to base-stations by flooding it to all discovered neighbours. This enables base-stations to collect data from nodes that have not been in radio range, but it can also lead to high demands for bandwidth, memory and energy. The second protocol is a history-based one, aiming to address the issues with the flooding protocol. Instead of flooding the data to all neighbours, it selects which nodes to send data to, based on prior communication patterns. The protocols were tested in simulation, using the ZNetSim simulator. For reducing power consumption, a very rough synchronisation method is used, having the radio communication scheduled for 30 minutes intervals that occur every two hours.

Anthony et al. [21] present a highly scalable solution for a five-to-seven year deployment for tracking migratory birds (Whooping Cranes), tagged with sensor nodes in the form of a backpack weighing less than 120 grams. The solution relies on a hybrid architecture that uses cellular networks for long-range communication and ad-hoc networks for short-range (in breeding and nesting grounds). The main difference between this scenario and the one consisting of tracking and monitoring wild horses presented in this thesis, is the requirement of the rate of GPS position samples (two per day as opposed to approximately forty-eight per day). Other differences consist of gathering other sensor data during the deployment besides the location information for monitoring the behaviour of the animals, in the case of the horse deployment.

We require the animals to be monitored in the wild for a period long enough to account for seasonal variation, this being over six months, but ideally aiming for twelve months. During this period we wish to record their positions (along with other sensor data) once approximately every twenty minutes, and the collected sensor information needs to be communicated to the nearest base-station. This poses a greater challenge compared to the applications described previously. Since all animal tracking deployments have the main goal to determine the animals' locations, which represents the most energy consuming information to acquire compared to other sensor data, a good metric to consider is the number of GPS positions sampled during the entire length of the deployment. For example, the deployment for tracking migratory birds [21] is estimated to have a lifetime of five-to-seven years, but they only acquire two GPS positions per day. This adds up to 3650 (five years) or to 5110 (seven years) samples, which is about 5.5 times less than our goal for the chosen scenario, which ideally requires 23890 GPS position samples (one sample approximately every 20-22 minutes over twelve months).

Most of the solutions presented for wildlife tracking / monitoring or node localisation applications using WSNs rely on frequent exchanges of radio beacons or control packets to establish links between nodes with the purpose of uploading the collected data [45, 40, 20, 49]. This has significant implications on the power consumption. For many WSNs applications, the option of replacing the battery once the nodes are deployed does not exist. Thus, conserving the energy is very important, and the best way to do it is to have the nodes sleeping as much as possible. However, the challenge is to wake them up at the same time, so as to exchange information.

The size and weight of the devices and the number of GPS positions over the entire deployment length are two relevant metrics for evaluating some of these deployments. The weight of the mobile sensor nodes is important for two reasons. Firstly, for wildlife tracking and monitoring applications, the devices that get attached to the animals should not influence the animals' natural behaviour, this imposing strict restrictions on the size and weight of the devices relative to the animals' sizes. Secondly, the size and weight of the node can be related to the efficiency of the algorithms running on the node. If a device can be heavy that means more battery power can be provided in the casing, thus compensating for the use of power-inefficient algorithms. It is relevant to consider the number of GPS positions sampled during the entire length of the deployment as a relevant metric, as the GPS module is usually the most power hungry component on platforms designed for tracking applications.

The solutions presented in papers [20] and [21] have an advantage over the solution proposed in this thesis, the long-range communication for the data upload. However, this advantage comes with considerable increase in power consumption, which leads to some of their weaknesses presented in Table 2.2.

Table 2.1: Summary of the Main Characteristics for the Most Relevant Animal Deployments

Paper	**Characteristics**								
	Description	Hardware weight	Localization / Other sensors	Accuracy	Deployment Duration	Network Protocols	Algorithm	Solar panel	Area covered
[19]	Monitored cougars	About 1kg	GPS	200m	6-8 months	No	-	No	2120km^2
[40]	Monitored European badgers	-	RFID collars / Temperature, humidity	30m	1 year	Collection protocol over XMAC	-	No	-
[20]	Monitored zebras. Algorithms tested in simulator	1.151kg	GPS	50m	1 year	Collection protocol (peer-to-peer network)	Flooding and History-based protocols	Yes	Thou-sands of km^2
[21]	Monitored migration of birds (Whooping Cranes)	<110g	GPS / GSM, 3D accelerometer, magnetometer, temperature	10-25m	5-7 years	GSM data upload and ad-hoc networks	-	Yes	>4000km
[15]	Tracked sheep	-	GPS	-	-	GSM + Cluster based approach	Distributed and centralized GSM	No	-
[22]	Monitored monkeys	N/A	Sound, video / Temp, velocity, humidity, light, wind, SO2, CO2	-	>1 year	Collection protocol	-	Yes	Qiling Mountain China
[14]	Underground tracking of badgers	105g	Magneto-inductive tracking	0.5m	>3 months	Contiki RIME over XMAC	-	No	-

Table 2.2: Strengths and Weaknesses of Previous Animal Deployments

Paper	Weaknesses	Strengths
[19] Cougars	Heavy: about one kg; No access to data during the deployment; Only six GPS positions acquired per day; Low positional accuracy (200 m); High infrastructure cost	-
[40] Badgers	The solution does not scale with the size of the animals' habitat; The frequency of the position samples is not controlled; The position accuracy depends on the no. of detection nodes deployed	Deployment length: one year
[20] Zebras	Very heavy: 1.151kg; Too dependent on the solar panel: lasts for only five days without solar charge	Deployment length: one year; Two radios, one for long range (8km); Nodes form a multi-hop network
[21] Birds	Only two GPS position samples per day (representing an average of 4380 of GPS positions samples during the deployment, which is more than three times less than the requirement for the scenario proposed in this thesis)	Deployment length: five-seven years; Uses GSM: data upload can always be possible if there is cellular network coverage; Good positional accuracy: 10-25m; Scalable solution
[15] Sheep	Determines group position, not individual positions; The algorithms presented have low synchronisation accuracy	-
[22] Monkeys	The solution proposed might be suitable only for monkeys; High infrastructure cost	Good deployment length: about one year; No devices attached to animals
[14] Badgers	Short deployment length: three months; The solution is not scalable	It is one of the few solution for monitoring animals underground

2.2.1.3 Hardware platforms used for animal tracking

This section presents some of the widely used, multi-purpose, hardware platforms targeted at WSNs applications. Benny Lo et al. present in [64] a wireless sensor platform for pervasive healthcare monitoring called BSN node, which uses the Texas Instrument MSP430 16-bit RISC processor (60KB+256B Flash memory, 2KB RAM), a wireless module with a 250kbps throughput and 50m range, and a 512KB serial flash memory. The chosen microcontroller requires 1.3mA during intensive computations and only 0.01mA in active mode. (In contrast, the EFM32GG990 used by our platform has a sleep mode with RTC with 32.768 kHz oscillator, Power-on Reset, Brown-out Detector, RAM and CPU retention using just 1.1μA, and 200μA/MHz @ 3V Run Mode). The platform is square-shaped with the length of 26mm and it can be integrated to use various sensors. It runs TinyOS and it can interoperate with other sensor platforms such as Telos and MicaZ. One of the earliest platforms, the Mica wireless platform [65] uses the 8-bit ATMega103L/ATMega128 microcontroller, which runs at 4MHz. It also uses a coprocessor, the Atmel AT90LS2343 microcontroller, to wirelessly reprogram the board and a 4Mb flash chip for persistent storage. The size of the board is 3.17x5.71cm, it uses inexpensive AA batteries and it runs the TinyOS operating system. The chosen radio module can support communication rates of 115kb and it can provide a range of approximately 60m. As the Mica platform is more useful for development, but less adequate for deployments, the Telos platform [66] was developed. It uses the MS430 microcontroller, the Chipcon CC2420 2.4GHz radio chip with a communication rate of 250kbps and an internal 2.4GHz Planar Inverted Folded Antenna. Both platforms Mica and Telos, were used to further develop wireless sensor nodes for various applications (e.g. the ActiS node presented by Milenkovic et al. in [67]). T. Choudhury et al. [68] present MSP (Mobile Sensing Platform), a platform used as part of an automatic activity recognition system that uses on-body sensor nodes. MSP was designed as a multimodal sensor board, allowing simultaneous data capturing from up to seven sensors. It uses the ATMega128 microcontroller; it has a miniSD card slot, Bluetooth radio, and a Li-Polymer battery of up to 1800mAh. As part of the activity recognition system, the MSP board is attached to an iMote, a 32-bit ARM7-based wireless node from Intel.

None of the platforms mentioned were designed with the power consumption features required for extended periods of deployment. It was therefore necessary to custom design the Prospeckz-5 specifically for this project.

2.2.2 MAC protocols

This thesis considers two protocols for the transfer of data from the mobile platforms to the base-stations. The first protocol is an ad-hoc asynchronous data transfer protocol with acknowledgement and redundancy presented in Section 3.1.1. This protocol is simple and deemed appropriate in a deployment scenario which requires low data rates and where the base-stations are mains-powered, and therefore can be kept on at all times. The second one is a synchronous protocol developed specifically for this class of applications involving tracking using GPS, called the Virtual-Beacon TDMA (VB-TDMA) protocol (presented in Section 3.1.2).

The proposed VB-TDMA algorithm is potentially more power efficient than existing MAC protocols for applications involving tracking mobile entities using GPS. It optimises the radio usage to the extent of using it solely for exchanging data packets, without any communications overhead. This is possible by using the GPS time data to generate an internal virtual beacon for precise synchronisation (order of milliseconds). This can be used by all the nodes to synchronise without being in range of any other node, and minimises the time during which the radio is turned on. Also, obtaining the time information from the GPS module does not imply any extra cost in terms of battery consumption, as this information is acquired along with the position coordinates which are essential requirements for the targeted class of applications.

In order to extend the battery lifetime for both the stationary and mobile nodes in the network, it is necessary to keep the components on the hardware platform in sleep mode for as long as possible, and wake them up simultaneously. Due to clock drift, it is not always possible to accurately synchronise the nodes based on their internal clocks during a long-term deployment. Also, in the case of a mobile ad-hoc network in which the nodes are not all in range of each other, the overhead of synchronisation is high, which is counter to the stated objective of keeping the power consumption as low as possible.

The VB-TDMA algorithm is different from other centralised and decentralised MAC protocols. Time Division Multiple Access (TDMA) protocols are part of the centralised category, where time is divided into slots to allow multiple node communications without causing collisions. For power-constrained devices, such as wireless sensor networks, the nodes are kept in sleep mode for as long as possible, and are awake only for the time when they are assigned a communication slot. Since accurate time synchronisation is required, traditional TDMA protocols have a base-station responsible for the coordination, which consumes more power than the other nodes in the network. This is not desirable in many WSN applications, where the nodes have limited energy. Also, in scenarios where nodes can be out of range of the beacon-generating node for extended periods of time, the communication overhead becomes high. Based on the level of uncertainty of the time synchronisation, the listening window has to be increased, which means that the power overhead increases significantly. An alternative to increasing the window size is having the desynchronized nodes convert to a non-synchronized method. The VB-TDMA eliminates the need for the base-stations to coordinate the synchronisation of all the nodes, and any exchange of control packets.

Decentralised MAC protocols also do not require a coordinating node in the network, eliminating the problem of unequal power consumption for the nodes. According to [69], this category can be divided into scheduled (e.g. S-MAC [70], T-MAC [71], LMAC [72]) and random-access protocols (e.g. B-MAC [73]). The scheduled ones use time slots similar to TDMA protocols. Their disadvantages lie in having larger time slots, having each node take on extra functionality by keeping a schedule of its time slots, and having the nodes periodically transmitting their schedule information to the other nodes in the network. These disadvantages translate to significant power consumption overhead for long-term deployments of scattered nodes. The low-power random-access protocols, such as B-MAC, SpeckMAC-B, SpeckMAC-D [69], WiseMAC [74] and X-MAC [75], intended for use in WSNs, do not use any schedule or synchronisation. However, they use in-channel sampling. They periodically listen for channel activity, turning on the receiver if activity is sensed, and turning it off either after receiving a packet or after a timeout.

There are many proposed TDMA protocols that use the global time of the GPS for synchronisation, the majority being specifically designed for vehicular networks [76, 77, 78, 79, 80, 81, 82, 83]. Conserving energy does not represent a priority for these protocols, as their main focus lies in reducing the communication delays. Depending on their targeted application scenarios, MACs for vehicular networks may have different goals: STDMA [76] focuses on maritime collision avoidance, VeSO-MAC [77] focuses on improved data throughput, MCTRP [78] aims for decreasing emergency message latency and increasing the network throughput, CBM-MAC [79] strives for guaranteeing safety message delivery with non-critical data support, and DM-MAC [80] seeks to provide collision-free and delay bounded safety message delivery with adaptable data throughput.

Ideally, TDMA-based MAC protocols aim to build a global schedule offering exact send and receive times for every node, thus completely avoiding the problems of overhearing and idle listening. In real-world scenarios, clock drifts and frequently changing external conditions render plain TDMA costly, as maintaining an accurate schedule is a complex and energy consuming task. Thus, in the case of TDMA protocols, aiming to minimise the energy consumption translates into attempting to reduce idle listening and overhearing [84].

The early MAC protocols were designed wtih energy efficiency as the main concern. However, more recent research (2007-2011) has moved towards providing multitask support and efficient delivery of bursty traffic, thus prioritising the throughput and delay metrics over power consumption. Huang et. al. [85] present the evolution of the design of MAC protocols for WSNs from 2002 to 2011. They do not only offer a classification of different MAC protocols, but present them in the way they naturally evolved from the problems that needed to be solved, changing their focus from power consumption to throughput and delay. Several multi-channel MAC protocols were proposed such as Y-MAC [86], MC-LMAC [87], MuChMAC [88]; however, the majority of them are less energy efficient than single-channel MACs under light traffic conditions. Incel et. al. [87] present MC-LMAC, a protocol that primarily focuses on maximizing the throughput by using parallel transmissions on multiple channels, thus avoiding interferences and contention. The sender nodes are allocated time slots which are assigned to specific channels. Kim et. al. [86] present Y-MAC, a multi-channel protocol that aims to achieve both high performance and low energy consumption under diverse traffic conditions. However, the Y-MAC performs better in terms of these two metrics under high traffic conditions. Crankshaft [89] is a single-channel MAC protocol for dense wireless sensor networks that aims to achieve high delivery ratios while having low power consumption. Crankshaft is designed for dense WSNs and its energy consumption overhead comes from employing node synchronisation and channel polling techniques. Tang et. al. [90] present PW-MAC, a protocol based on asynchronous duty cycling that aims to reduce the power consumption by having sender nodes predicting the receivers' wakeup times. PW-MAC uses an on-demand prediction error correction mechanism and a prediction-based retransmission mechanism for addressing time challenges and maintaining energy efficiency while collisions occur.

Since the class of applications tackled by this thesis has energy consumption as the primary concern, the comparison focus of the VB-TDMA protocol is against the other MAC protocols that are most suited for this class of applications, the ones that also have as a primary aim to minimise energy consumption.

2.2.3 Hardware simulation models

This section presents an overview of previous work related to hardware simulation models for wireless sensor networks.

Shnayder et al. [91] present PowerTOSSIM, an extension to the TOSSIM simulator, which supports large WSNs simulations and provides accurate per-node estimates of power consumption. It estimates the number of CPU cycles executed by each node, without requiring instruction-level simulation. The hardware energy consumption model is based on the Mica2 platform, a newer generation of the MICA Mote [65]. The authors validate the accuracy of the power consumption estimates against fifteen TinyOS applications, obtaining an arguably low percentage of 0.45% to 13% of the true power consumption of nodes running an identical program.

Mallanda et al. [92] present results obtained with OMNeT++, a WSNs simulator based on the discrete event simulation framework. They validate the simulator by comparing its results to the ns-2 simulator [93] and state that the performance of OMNeT++ is superior to that of the ns-2 in terms of execution speed, memory usage, and ease of modifying the network. The simulator is described as coming with basic models for simulating a battery, CPU and radio.

Other existing implementations of basic battery and radio modules are presented in [94], [95]. Park et al. [96] present features added to the ns-2 simulator, that enable the nodes in the simulations to have a more detailed power model corresponding to the underlying energy-producing and energy-consuming hardware components. It consists of a battery model that provides energy to consumer models such as the Radio, CPU and sensors. Three types of battery models that resemble real battery behaviour are proposed, and can be used to study how different battery behaviours can affect the energy efficiency of applications: a linear model, a discharge rate dependent model and a relaxation model.

We have similarly modelled a hardware platform in a simulator for wireless sensor networks. The differences include the use of a significantly more recent design, that of the Prospeckz-5 (designed in 2013 compared to platforms such as Mica2 designed in 2004), which makes use of newer-generation components offering better energy efficiency. The simulations presented in this thesis were performed in the SpeckSim simulator introduced in Section 2.1.4. The simulator was extended with models of the new Prospeckz-5 hardware platform, which was validated using data from a deployment, as presented in Section 5.1. The models for the hardware platform's components are described in Section 3.2.1, and their main aim is to accurately estimate the energy usage of the battery operated devices that use the Prospeckz-5.

2.2.4 Mobility models

This section presents an overview of previous work related to mobility models for ad hoc and cellular networks.

ZebraNet [20] represents a thirty-node deployment with the purpose of tracking zebras in Kenya. The experimental results presented in the paper are gathered from simulation. Three possible scenarios were considered: an ideal scenario, a storage-constrained scenario and a bandwidth-constrained scenario, for three proposed protocols, looking at two metrics: upload success rate and energy consumption. A simulation environment to resemble ZebraNet and a mobility model for zebras based on observational studies were built in the ZNetSim simulator. Predictions of zebras' meetings are made based on the mobility model, which itself is based on estimations (random direction, estimated speeds) and assumptions (how many times per day the zebras go to a water point and to which one). In comparison, the data-mirror mobility model proposed in this thesis in Section 5.1.2 is using only real data with a twenty-minute resolution and less than ten-metre accuracy.

Mobility models for ad hoc wireless sensor networks can be broadly classified as purely synthetic, domain-driven and data-driven. Other classifications in literature are mostly categorizing them as synthetic and trace-based.

The importance of mobility models for the performance evaluation of network protocols is argued throughout literature [97, 98, 99, 100, 101, 102, 103]. Many publications survey mobility models for ad hoc and cellular networks, arguing the benefits of both synthetic and trace-based models. Synthetic mobility models are important due to the level of difficulty of modelling new network environments by trace-based models if traces have not yet been created [99]. However, simulation results of synthetic mobility models, which are based on randomly generated movement that create synthetic traces, often do not match realistic scenarios. Trace-based mobility models, which are based on traces captured from the real world, are more realistic and provide higher accuracy movement patterns.

Aschenbruck et al. [97] highlight that the accuracy of a mobility model matters if it makes a difference in its impact on the results in the performance evaluation with the model. If there is no impact on the performance evaluation results, then there is little use of a more detailed model. [97] summarizes the research conducted in the area of mobility modelling, concentrating on trace-based and synthetic mobility models, and [97, 98] discuss the process of collecting traces from real scenarios. Tuduce and Gross [98] propose a structured framework for extracting mobility characteristics from a WLAN trace and determining a set of parameters that can then be used to generate mobility scenarios for simulations. Cross validation was used to validate the model distributions, and the model was compared using the IMPORTANT framework [104] against two other models: Random Waypoint [105] and Reference Point Group Model [100].

Camp et al. [99] classify mobility models based on a second criterion: entity and group mobility models. The paper discusses several synthetic entity mobility models such as: Random Walk, Random Waypoint, Random Direction, Gauss-Markov and City Section mobility models; and several group mobility models as well, such as: Exponential Correlated Random Mobility Model, Column Mobility Model, Nomadic, Pursue and Reference Group Mobility Model. Hong et al. [100], present a survey of mobility models for both cellular and multi-hop networks, focusing on the Random Walk mobility model. They also propose a new group mobility model (the Reference Point Group Mobility model) and investigate its impact on specific network protocols. Hsu et al. [102], propose a framework for evaluating the performance of ad hoc networks called Weighted Way Point mobility model. This proposed model offers more realistic performance evaluation than the Random Waypoint model, as its nodes display uneven (clustering), time-varying spatial distribution. [101] proposes an enhanced random mobility model, which correlates the new values for speeds and directions to previous ones, and uses an acceleration to achieve its new-targeted speed.

This thesis proposes the data-mirror mobility model (presented in Section 5.1.2), which uses a slice of real movement data that should preferably be as representative as possible for the entity's movement patterns. The more representative the slice, the higher the accuracy offered by this model. It essentially uses the same paths from the provided slice of movement data by making the mobile nodes go back and forth on them. This data-driven mobility model provides higher accuracy in predicting the network behaviour when compared to trace-driven ones for the movement of wild horses, as shown in Section 5.1.2.

2.3 Summary

This chapter has presented the application scenarios implemented in this thesis, the custom-designed hardware platform and the chosen simulator for analysing the behaviour of sensor networks. It has given an overview of existing work and research in the areas touched by this thesis, discussing their weaknesses and shortfalls, and how these are addressed by the solutions proposed. The weaknesses and shortfalls of the related work confirm the existence and challenging nature of the identified class of applications of long-term tracking and monitoring of mobile entities in the outdoors.

Taking this into account, the next chapter presents the proposed data upload protocols that overcome these challenges for this class of applications. It also introduces the implementation details of simulation models developed to evaluate the performance of these protocols and compare them to the appropriate alternatives.

Chapter 3

Communication Protocols and Simulation Models

Long-term deployments are challenging as they allow few iterations, if at all, for testing, experimenting and incrementally improving the solutions. This motivates the need for the development and use of simulation tools, for predicting the results of such deployments. This chapter presents the core of our solution for the targeted class of problems, the data upload Virtual Beacon TDMA protocol. It also presents the evaluation environment developed for validating and characterizing this solution, consisting of the implementation of the hardware platform simulation models and the mobility model in the SpeckSim simulator. It addition to this, a basic asynchronous protocol is proposed as a simple alternative in certain scenarios that benefit from mains-powered base-stations. The chapter concludes with the results of a simulation of tracking and monitoring horses in the wild, based exclusively on the parameters from the datasheets of the hardware components, and therefore without the benefit of data input from deployments.

3.1 Communication Protocols

Two communication protocols are presented: a synchronised Virtual-Beacon TDMA algorithm for scenarios with power-constrained base-stations, and a basic asynchronous data upload protocol with redundancy targeted only at scenarios with mains-powered base-stations.

These are two distinct designs, both of which have been tested in the field on a representative use case for the chosen class of applications, the long-term tracking and monitoring of horses in the wild. The asynchronous protocol was tested in a long-term deployment on thirty-two wild horses, whereas the VB-TDMA was employed in a small experimental deployment of eight domesticated horses. Both protocols were evaluated and their performances compared in simulations.

The aims of the design of the communication protocols are to minimise the power consumption so as to extend the battery life, and to promote simplicity leading to robustness which is necessary for successful long-term deployments.

3.1.1 Asynchronous Data Upload Protocol with Redundancy

For the particular scenario of the horse tracking and monitoring deployment that took place in Doñana National Park, Spain, the base-station nodes were placed on selected towers within the nature reserve to gain better radio range and, importantly, for access to power and network connectivity. The access to a power supply enabled the base-stations to keep the radios switched on, listening for packets at all times. Batteries were used as a backup solution providing redundancy in case of brief power drop-offs. It is important that the ellipsoid radio ranges of any two base-stations do not intersect, to avoid collisions of acknowledgement packets. Therefore, to allow deployment of the base-stations at the fixed locations of the towers, adjacent base-stations used different radio channels, operating on different frequencies.

The asynchronous data upload protocol was designed to accommodate this scenario, along with other use-cases, such as mobile base-stations used by biologists as they drive around, or attached to UGVs (Unmanned Ground Vehicle) or UAVs (Unmanned Aerial Vehicle), for alternative forms of data collection.

The mobile nodes are programmed to upload data onto three different channels every fifteen seconds. Even though when transmitting (TX mode) the radio and the amplifier together draw 28.3mA, radio transmissions generally use very little power, as the time required to send a packet is very short (less than 1ms). The data packet size is 32 bytes. In contrast, when the radio is used to listen for packets (RX mode), it is turned on for longer periods of time, during which it draws 13.5mA, and considering the energy consumption of the radio amplifier, the total consumption is of 23.5mA. Using this low-power protocol for this application, results in the GPS power consumption dominating all other components, representing more than 95% of the total power consumption. This suggests that the frequent number of radio transmissions, such as one every fifteen seconds, does not have a significant impact on the overall power consumption of the system. However, it does increase the responsiveness of the network, allowing us to better take advantage of the brief node meeting times in the data upload process.

The nodes also have a fast upload channel, separate from the other three channels, to accommodate mobile base-stations (using UAVs and UGVs) use-cases. On this channel they upload data in short bursts of up to six packets (for each transmitted packet an acknowledgement needs to be received before transmitting the next packet).

Using different channels offers redundancy in the data collection process, as on each channel the complete set of data collected by the mobile nodes is uploaded. In other words, if the data collected by one base-station is lost, that data may be found at one of the other base-stations that are programmed to listen on a different channel. In challenging deployments such as this one, out in a wildlife reserve, it is beneficial to have redundancy for reducing the data loss, as the base-stations are liable to lose power or network connectivity. The mobile nodes using this protocol are configured to transmit packets in chronological order (from the oldest to the newest), and to resend them until they receive an acknowledgement message.

3.1.1.1 Implementation

This basic asynchronous protocol was influenced by the need to conserve the battery on the mobile nodes for long-term deployments (over six months, preferably twelve months), given that the base-stations had access to power. The mobile sensor platforms are designed to upload a complete set of data on four different channels: three main channels and a fast upload channel where packets are uploaded in bursts of six. This enables adjacent base-stations to be programmed to listen on different channels, to avoid packet collisions. The range of the communication between a mobile node and a base-station is around 1000m. In the interest of reducing power consumption, a collisions avoiding algorithm was not implemented. Any packet collisions due to multiple mobile nodes sending packets simultaneously on the same channel to the same base-station are extremely rare, given the low density of mobile nodes deployed. In the case they do occur, the nodes back-off and resend.

Mobile Node. When a mobile node starts up, its radio is set to use the first of four channels. It starts a recurring timer with a delay of twenty minutes which, when triggered, wakes up the GPS module and waits for a position fix. After acquiring a fix, the GPS module is kept on for some extra time (usually three seconds, but for every fifth interval the extra time is nineteen seconds, to update the satellite information to improve the positional accuracy). When the GPS is kept on after getting a fix, its state changes from "Acquisition" to "Tracking", the former having slightly lower power consumption. When this process is completed, the GPS returns the position coordinates of the node, along with a time stamp, the time it took to acquire the fix and the extra time it was on after getting the fix. The same timer also delimits the activity and head orientation intervals, these values being calculated based on accelerometer data sampled throughout the twenty-minute intervals. However, in the case of simulations, it is only intended to simulate the power consumption of the accelerometer,without generating any sensor readings. A packet is composed of information provided by the GPS and the rest of the sensors (in simulations, the rest of the sensors' data is dummy data). The Prospeckz-5 implementation places the packet in a queue and allocates it four flags, indicating the channels on which it was uploaded. However, in simulation, the packet is placed in four queues, each corresponding to one of the radio channels, making it easier to study the queues' dynamics. The node starts another recurring timer with a delay of fifteen seconds, used for uploading data to base-stations. When this timer triggers, if the queues assigned to the four channels contain packets, one packet out of each queue is sent on its corresponding channel. A mobile node can have up to fifteen retransmissions of a data packet if an acknowledgement message is not received, in which case, after fifteen seconds it will retry uploading the same packet. Nodes always try to upload the oldest packets from the queues, and these are not removed from the queues until acknowledgements are received. The NRF24L01+ radio on the mobile nodes uses its specific automatic acknowledgements option, which briefly switches the radio to "Listen" mode after every packet transmission to receive a short and quick acknowledgement message. This option is implemented in hardware to offer a quick, low-power acknowledgement system.

Base-station node. At start-up, a base-station is assigned a channel to listen on and sets its radio to "Listen" mode. The received data is either stored locally or forwarded, depending on the application. In simulations, the base-stations filter duplicate packets (which may occur due to loss of acknowledgement messages) by keeping a "dictionary" with node identifiers as keys, and lists of at most fifty data packets as values, each list corresponding to a mobile node. When a base-station receives a data packet, it immediately triggers an acknowledgement message, then checks if the packet is a duplicate. If not, the packet is stored (written to an output file) and added to the corresponding duplicate filtering list.

3.1.2 Virtual-Beacon TDMA

The VB-TDMA algorithm was designed to be a low-power communication protocol for tracking applications and requires a hardware platform that contains a GPS module. TDMA protocols guarantee a time slot for each sender in a network, by relying on an accurate clock to synchronise communications between senders and receivers. When tracking mobile entities in wide geographical areas (tens to hundreds of square kilometres), their positions need to be marked with time stamps provided by a global clock. A GPS module provides both the location information and a precise global time, which can be used to synchronise the communication of the nodes in the network.

The synchronisation of the nodes based on the GPS time information works as follows. The GPS module supports two protocols for communication: NMEA and SiRF. The NMEA protocol does not provide access to any of the GPS's power modes and the GPS module in SiRF mode is not able to get 1PPS time messages. 1PPS time messages carry the time associated with the current 1PPS pulse. Each message is output within a few milliseconds of the 1PPS pulse and contains the time of the pulse with microsecond accuracy. Since both these aspects are critical to the application, it was necessary to use both protocols: SiRF was used to set the GPS module in the Micro Power Mode, and NMEA to activate the ZDA messages, which enable the 1PPS time messages.

The proposed TDMA algorithm uses two types of time slots: Discovery slots and Upload slots. The Discovery slot has the purpose of determining if a certain mobile node is within range of a base-station. A mobile node sends only one packet in this slot, and if it receives an acknowledgement and if it has data to upload, it schedules a larger slot, called an Upload slot. If not, it goes into sleep mode and wakes up for the next Discovery slot. The base-station has the radio turned on in receive mode for the period of all the Discovery slots. If during one of the slots it receives a packet, it acknowledges that packet and schedules to turn on the radio for the corresponding Upload slot. If a mobile node does not manage to upload all of its data in the first Upload slot (and it does not go out-of-range of the base-station during the time of the Upload slot), another Upload slot is scheduled. This process is illustrated in Figure 3.1.

The radio on the mobile node can be configured to have a number of retransmissions of a packet in case it does not receive an acknowledgement. This number is chosen to suit the communication environment where the deployment will take place,

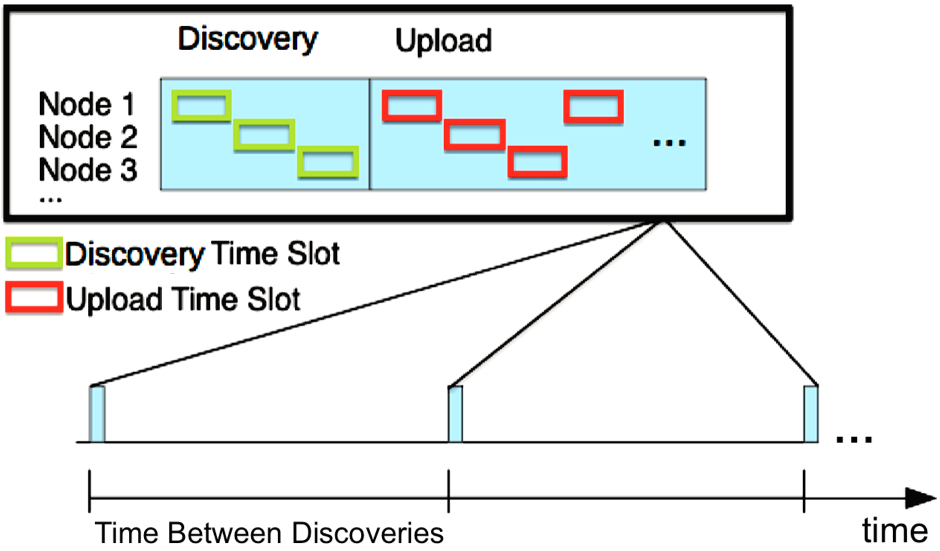


Figure 3.1: VB-TDMA algorithm.

and is also reflected in the size of the Discovery slots. For example, fifteen retransmissions were timed to be slightly greater than eight milliseconds, therefore an appropriate Discovery slot could be ten milliseconds.

The pseudo-code in Figure 3.2 presents only the core functionality of the VB-TDMA protocol, avoiding the complexity of other features such as multichannel upload, dynamic allocation of upload slots or multihop functionality.

Multiple node discoveries can be scheduled during the time interval between data collection occurrences, which in the case of the horse tracking scenario is approximately twenty minutes. Frequent node discoveries increase the responsiveness of the network, which could make a difference in the data upload process for scenarios with short node meeting times. Discovery phases do not have a huge impact on the overall power consumption, as the power consumption overhead of the radio transmitting a packet is almost negligible compared to the GPS's energy consumption when acquiring a location. The VB-TDMA was also designed to offer the functionality of uploading complete sets of data on multiple channels.

In simulations, the data collection process for the mobile nodes consists of sampling the GPS sensor, which returns the position and time, along with sampling the accelerometer and light sensors, which only discharge the battery accordingly.

```
Perform initial GPS configuration
Acquire an initial GPS fix
Synchronize clocks
Schedule periodic "GPS Sampling" and "Discovery Phase"

If mobile node
  • On "GPS Sampling"
      o Acquire GPS fix
      o Synchronize internal clocks
      o Sample sensors / process sensor data
      o Build packet with sensor data
      o Store packet in flash
      o Go to Sleep Mode
  • On "Discovery Phase"
      o Transmit one packet
          ▪ If (ACK received)
              • If (more packets to upload)
                  o Schedule "Upload Phase"
              • Mark packet as delivered
          ▪ Else
              • Retransmit up to X times
      o Go to Sleep Mode
  • On "Upload Phase"
      o Start Timer
      o While (packets to upload)
          ▪ Transmit packet
              • If (ACK received)
                  o Mark packet as delivered
              • Else
                  o Retransmit up to X times
                  o If (unsuccessful)
                      ▪ Stop Timer
                      ▪ Go To Sleep Mode
      o On "Timer triggered"
          ▪ Stop transmissions
          ▪ If (packets to upload)
              • Schedule "Upload Phase"
          ▪ Go to Sleep Mode

If base-station
  • On "GPS Sampling"
      o Acquire GPS fix
      o Synchronize internal clocks
      o Go to Sleep Mode
  • On "Discovery Phase"
      o Start Timer
      o Listen for packets
      o On "packet received in slot X"
          ▪ Send ACK
          ▪ Store packet
          ▪ Schedule the "Upload Slot" for node X
      o On "Timer triggered"
          ▪ Stop listening
          ▪ Go to Sleep Mode
  • On "Upload Phase"
      o Start Timer UploadPhase
      o Start Timer OutOfRange
      o Listen for packets
      o On "packet received"
          ▪ Send ACK
          ▪ Store packet
          ▪ Reset Timer OutOfRange
      o On "Timer OutOfRange triggered"
          ▪ Stop listening
          ▪ Stop Timer UploadPhase
          ▪ Go to Sleep Mode
      o On "Timer UploadPhase triggered"
          ▪ Stop listening
          ▪ Stop Timer OutOfRange
          ▪ Schedule the next "Upload Slot" for this node
          ▪ Go to Sleep Mode
```

Figure 3.2: Pseudo-code of the core functionality of the VB-TDMA protocol.

3.1.2.1 Extensibility and Scalability - Hardcoded Discovery, Dynamic Upload

In addition to prioritising energy efficiency, scalability in terms of network size, node density and topology, is also important. The protocol should be flexible to accommodate various network changes that may occur, including network growth. Depending on the application, nodes may die over time, new nodes may join the network later on, and all of them are continuously moving and changing their locations. A suitable protocol should handle such network changes in a capable manner. Many metrics are considered important for MAC protocols, such as fairness, latency, bandwidth utilisation and throughput. These are usually the primary concerns in traditional wireless voice and data networks, but they are secondary in wireless sensor networks, including in our targeted class of applications.

When designing protocols, it is always a matter of prioritising certain metrics of interest over the others. As one protocol cannot excel in every way, the solution comes from identifying the satisfactory trade-off between the metrics used for quantifying the protocol's performance, in order to best suit the target scenario or class of applications.

The VB-TDMA exploits the presence of GPS modules in the nodes used in this class of applications to eliminate the wastage of energy due to collisions, overhearing, control packet overhead and idle listening. Based on the application scenario, it further reduces the energy consumption by spacing out the Discovery phases, up to the point when the amount of data uploaded would be affected. For the applications within our target class, network latency is secondary to energy efficiency and the volume of data uploaded. The latency is also highly dependent on the movement of the mobile nodes, and maintaining its minimisation at a low priority does not affect the practicality of the solution.

Further we discuss the features that offer the VB-TDMA some flexibility to changes in the network size, and that make it scalable, enabling it to accommodate deployments of large networks (hundreds/thousands of mobile nodes). The simplest method for adding new mobile nodes to an ongoing deployment is to account for their slots beforehand, when programming the base-stations. This way, the base-stations will listen longer for packets during the Discovery phase to account for the extra slots. And each extra slot of the Discovery phase will have assigned a corresponding inactive Upload slot. Given the ten-millisecond Discovery slots, one extra second in the Discovery phase is enough to add one hundred new mobile nodes. For low numbers of mobile nodes, the power consumption impact of the Discovery phases on the base-stations due to the extra slots is reasonably low. However, the disadvantage is that it lengthens the Upload phase, thus making the upload process slower, even though many of the extra Upload slots are not used for most of the deployment, or may not be used at all. The Upload process is most inefficient when having a large deployment (in terms of the number of mobile nodes) and only a few mobile nodes in range of a base-station. Figure 3.3 shows that the upload time increases linearly with the deployment size for the extreme case of having only one mobile node (the one with the highest ID) with a significant amount of data to upload (needing ten upload slots for uploading all its data). The length of the Upload slot is considered to be 100ms in this example.

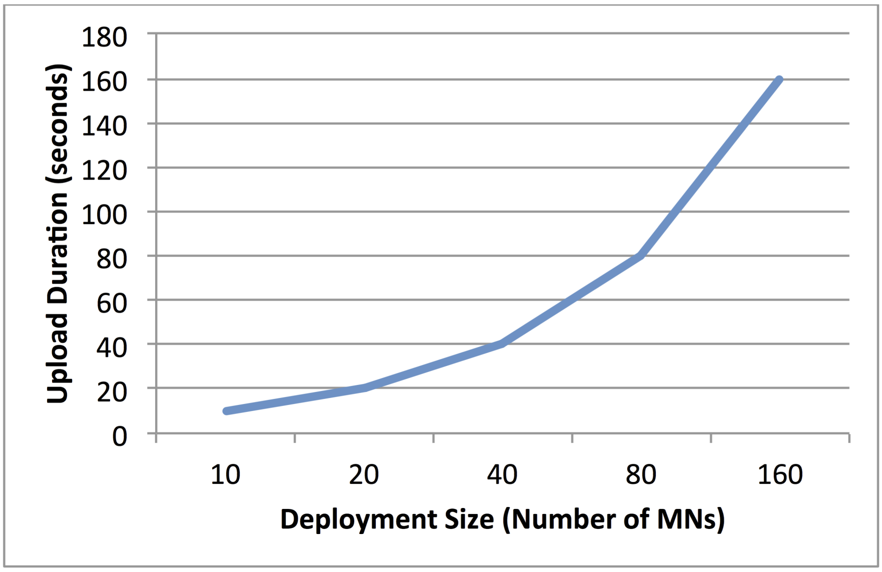


Figure 3.3: The upload duration when needing ten upload slots of the mobile node with the highest ID in the deployment.

For example, consider a deployment of fifty nodes, and one of the mobile nodes, MN50, needs ten upload slots for uploading all its data. It is in range with a base-station that has no other mobile nodes around it. Even though an Upload slot is 100ms, instead of taking one second for MN50 to upload all its data, due to all the inactive Upload slots of the nodes within the deployment, MN50's data upload will take fifty seconds.

This problem makes the solution highly unscalable, and it was addressed by introducing **dynamic allocation** of the Upload slots. The size of the base-stations' Discovery phase is kept the same, set accordingly to the maximum deployment estimated size expressed as the number of mobile nodes (e.g. 1 second = 100 mobile nodes for 10ms Discovery slots). During the Discovery phase, a mobile node sends one packet and the base-station receiving this packet replies with an acknowledgement (ACK) containing the number of the Upload slot allocated to this mobile node. The Upload slots are allocated continuously, in numerical order (e.g. if MN1 uploads a packet to BSX it gets Upload Slot 1, and if MN10 is the next mobile node that uploads a packet to this base-station, it gets Upload Slot 2). During the Upload phase, a base-station's ACK carries the number of mobile nodes in range with that base-station, which represents the total number of Upload slots for the current Upload phase. This enables the mobile nodes to schedule their following Upload slots. They activate timers from the start of the Discovery phase for counting the Upload slots. The process is outlined in Figure 3.4.

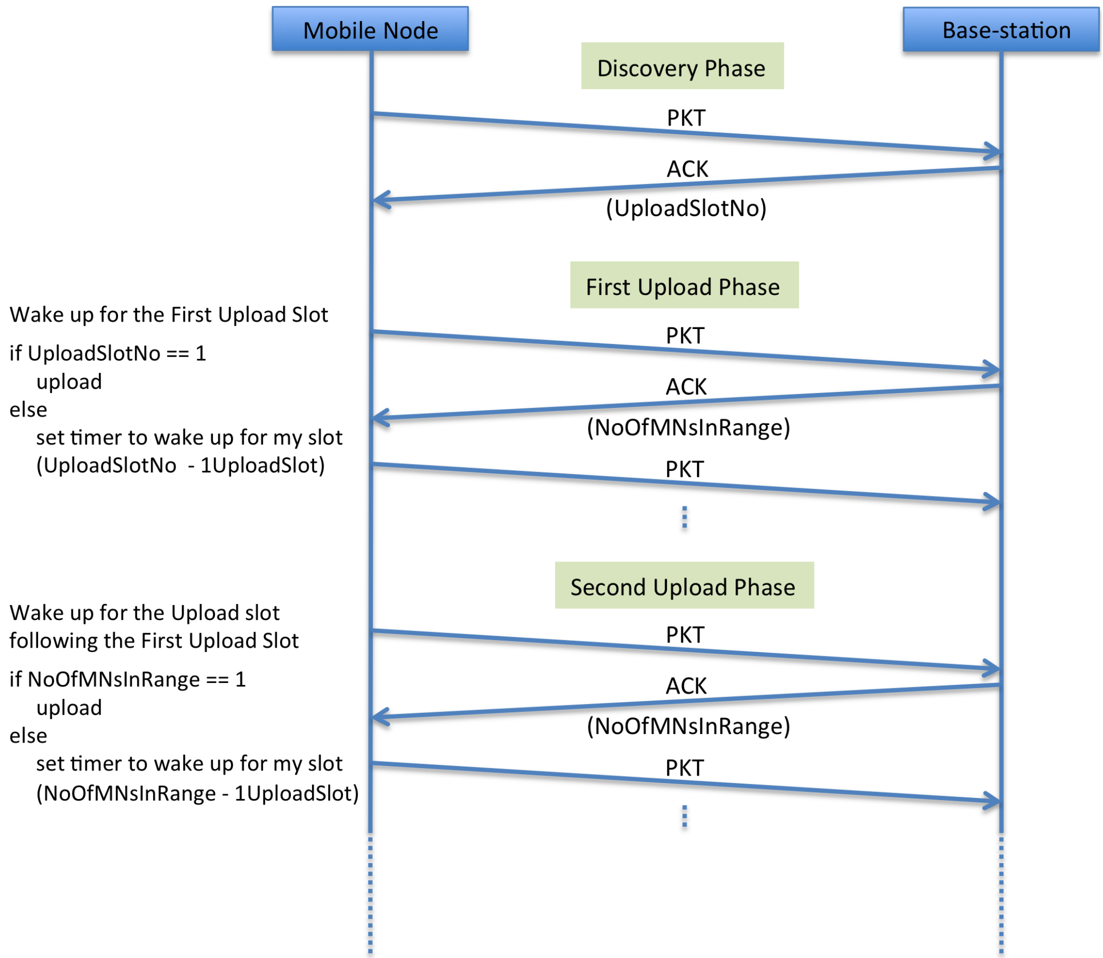


Figure 3.4: Dynamic upload process.

3.1.2.2 Multihop (Store-and-Forward) Functionality

The multihop functionality of the VB-TDMA enables mobile nodes to exchange data packets so those rarely or never in range of base-stations can upload their data via other mobile nodes that come in contact with them, and pass within range of the base-stations. Another two-phase (involving discovery and upload) timing scheme was created for exchanging packets between the mobile nodes, which alternate as senders and receivers depending on the time since their last contact with a base-station. In the case of the wild horses tracking scenario, the nodes that do not come in contact with a base-station for more than twenty-four hours become senders.

As opposed to the Discovery phases of the data upload process to base-stations, where base-stations offer each mobile node a time slot for sending data, the mobile node receivers offer a lower number of time slots than the number of mobile nodes in the deployment. The selection of this number is specific to the scenario, depending on the size of the deployment and spread of the nodes. For example, if this number is set to ten, a mobile node can collect data from a maximum of ten others during a data exchange following the Discovery phase. As sender nodes are aware of the time of the Discovery phases of the receivers, they randomly choose a slot on which they send one packet. If the packet gets acknowledged, both the sender and the receiver activate the corresponding Upload time slot, in a manner similar to the process of uploading to base-stations.

Flooding the network with redundant sets of data would have a negative impact on the power consumption and memory usage of the nodes. Packets that have been previously uploaded to base-stations, even though not on all channels, are not considered for upload to mobile nodes. Each node holds a HashSet with the sequence number of every packet uploaded to mobile nodes, in order to not upload the same packet to more than one other mobile node. The packets received by mobile nodes are stored in a separate queue, referred to as queueMN. In the event that a mobile node fills its memory, it will not accept packets from other mobile nodes, and always discards the oldest packet from queueMN when it collects more data itself. Unless the memory is full, packets are only discarded when uploaded to base-stations. When a node comes in range, it first uploads all of its data on the base-station's particular channel, and then the data collected from other mobile nodes. In the case of the Doñana deployment, it is highly unlikely that mobile nodes would run out of memory, as the flash chips on the Prospeckz-5 are sufficient to store approximately ten complete sets of data over the course of one year.

3.2 Simulation Models

The SpeckSim simulator, introduced in Section 2.1.4, was selected for its extensibility and collection of simulation models for radio communication that take into account key aspects such as frequency and phase modulation, noise and path loss.

As hardware characteristics of platforms needed to be captured in the simulation, separate layers were implemented from the ground up. "Devices" are the top layer of simulated entities and essentially model wireless sensor nodes. Each device object contains an instance of models of the hardware, software and communication protocols required by the simulated physical platforms. The main purpose of hardware component models is to simulate the energy usage of battery-operated sensor devices.

3.2.1 Hardware Platform Modules for Estimating Power Consumption

3.2.1.1 Microcontroller

The simulator includes models for a microcontroller (MCU) and clock system. The "MCU" model was designed to capture the current consumption of the main states of the Energy Micro EFM32 Giant Gecko microcontroller [106], and to be used by SpeckSim's power analysis module. The clock models aspects of the real-time clock in any microcontroller, enabling clock frequency drifts to be simulated. The clock also provides a real-time scheduling interface for simulated software components to schedule events.

3.2.1.2 Radio

The "Radio" models the Nordic NRF24L01+ radio chip along with the RFaxis RFX2401C radio amplifier in terms of power consumption and behaviour. Besides implementing the device's states in terms of current consumption, the radio model within SpeckSim is designed to also match characteristics such as: bitrate, receiver sensitivity, transmission power levels, use of different channels, and the behaviour of the quick and low-power auto-acknowledgements implemented in hardware, which are characteristic to the NRF24L01+. When a transmission is initiated, an event is posted in the simulator's queue that takes into account the transmission power level, bitrate, current consumption, the message size and interferences. The duration of the event is calculated based on the size of the message and the radio's bitrate. In SpeckSim the signal strength degrades with distance (and other environment obstacles such as barriers). Once a transmission is initiated, a Java map structure is created containing the received signal strength (dBm) for all the nodes in the simulation. This map consists of the transmitter node identifier, the receiver node identifier and the received signal strength. A node does not consider a transmission initiated by itself. However, all other nodes acknowledge detecting a transmission if its received signal strength is above the radio's receiver sensitivity. The received signal power is calculated based on diminishing the transmitter power level due to path loss which is dependant on the distance between the nodes. In addition to degrading the signal strength with distance, interferences are taken into account when calculating the signal-to-noise ratio. Interferences may occur when multiple nodes are transmitting simultaneously. In the case of detected transmissions, if the signal-to-noise ratio is not higher than a set threshold, the receivers ignore the transmissions, and the packet is considered lost. If the signal-to-noise ratio is sufficient for the radio to pick up a packet, other factors that can affect the packet's successful reception are then taken into account, such as probabilistic noise and packet loss (calculated according to the modulations schemes FSK and PSK). The signal-to-noise ratio threshold for reception is calculated based on the packet size and a configurable probability in the case of each modulation (FSK and PSK).

3.2.1.3 Sensor and GPS

The "Sensor" model was designed to reflect the power consumption of the Freescale MMA8451Q accelerometer and the Maxim Integrated MAX44007 light sensor. This model does not aim to simulate the actual sensor data for either of the sensors, as this is not of interest in the simulated experiments. The "GPS" model was designed to resemble the performance of the FGPMMOPA6H external GPS module. The performance of the simulation model in terms of the time it takes to acquire position fixes is modelled to reflect the performance of the real module in the environment of the deployment on wild horses. This is achieved by using a Gaussian distribution to generate random numbers. Its mean is the average time to get a fix of the real GPS modules from approximately 80000 data samples. The standard deviation was chosen to match the range of the times from the data collected in the actual deployment. The energy consumption for the different states of the GPS module was taken from its datasheet.

The model also permits injecting into SpeckSim the real time-to-fix data gathered in the deployment. The GPS position error was not modelled, as it is outside the scope of the simulated experiments.

3.2.1.4 Battery

Lithium Thionyl Chloride batteries combine excellent energy density with extended operational temperature range and minimal self-discharge, making them an excellent choice for equipment in long-term deployment scenarios. The "Battery" model was designed to resemble the discharge characteristics and current capacity of the Lithium Thionyl Chloride primary cells used on the mobile nodes, as stated in the manufacturer's technical specification. It was estimated that two such batteries (2x2500mAh) would allow operation of the platform on the mobile nodes for approximately one year.

3.2.2 The Mobility Model

Mobility models capture the movement of the mobile entities over time. The position and orientation of both static and mobile nodes at time instants are specified within the simulation environment, with the provision to set specific paths for the mobile nodes. A path consists of location points within the simulation area, and mobile nodes can move from one location point to the next at different speeds. The implementation of the model allows the injection of real data gathered from the deployment into the simulation in order to faithfully reconstruct the movement of the mobile entities. This permits accurate replication of the actual movement behaviour of the horses from the deployment, by mapping the nature reserve area into the simulation and matching the movement speeds of the horses between locations. This is achieved by converting the actual GPS coordinates to those within the simulation, and calculating average speeds for the simulated nodes based on the distances between successive GPS positions for consecutive time stamps.

A trace-based mobility model was implemented in the simulator, which generates new locations and speed values for nodes in a pseudo-random fashion. The new locations are based on the current location of the node and a newly generated distance. The new distance is generated randomly using a Gaussian distribution, with the mean the value of the average distance travelled by the horses in the actual deployment (using approximately 140,000 samples), and the standard deviation reflecting the spread of distances derived from the real data. This distance is used as the radius for a projected circle with the centre in the current location of the node. The next location is chosen randomly from the points on the circumference of this circle, while satisfying the condition of not exceeding the boundary of the area that confines the nodes. The speed of the node is generated in a similar fashion, also using a Gaussian distribution, with the mean being the average speed calculated from the data gathered from the real deployment, and the standard deviation matching the spread of the real speeds.

3.2.3 Import of Real Data

This section describes the process used for replicating the deployment of sensor nodes on the wild horses in the Doñana National Park, Spain, within the SpeckSim simulator.

The imported data is of two types:

- Constants calculated from a dataset of approximately 140,000 packets collected from the initial few months of the deployment, such as:
  - Average distance travelled by the horses in approximately twenty minutes: 0.099km
  - Average speed of the horses: 0.29km/h
  - Average time to fix for the GPS modules: 25.23s
  - The area where the horses are located:
    - North (maximum Latitude): 37.20561
    - South (minimum Latitude): 36.92878
    - East (minimum Longitude): -7.561832
    - West (maximum Longitude): -6.353243
- Input files for each node, containing real data:
  - Input files for the GPS module containing the time to acquire the positions. A file with data corresponding to nodes from the real deployment is generated for each simulated node.
  - Input files for the Movement model, that contain a node's next destination and its speed to that destination. Nodes' paths are determined by the next-destination values, which are obtained through the conversion of the GPS locations to locations within the simulator. The speed assigned to each next-destination value is also generated from the real data. It is calculated

using the distance between two GPS consecutive locations and the time it took a horse to travel from one to the other.

The area delimited by the horses movements was determined using the GPS locations with the highest and lowest values for the Latitude and Longitude of all locations gathered. These values are represented by numbers one to four in Figure 3.5. The quadrilateral defined by numbers five to eight represents the area where the horses were located. By eliminating six outliers (the locations in blue, including numbers one to four), the size of the area occupied by the horses is reduced 6.42 times. This smaller area is highlighted in green by numbers thirteen to sixteen in Figure 3.5, and its approximate dimensions are 10.57km by 10.44km (about 110km^2).

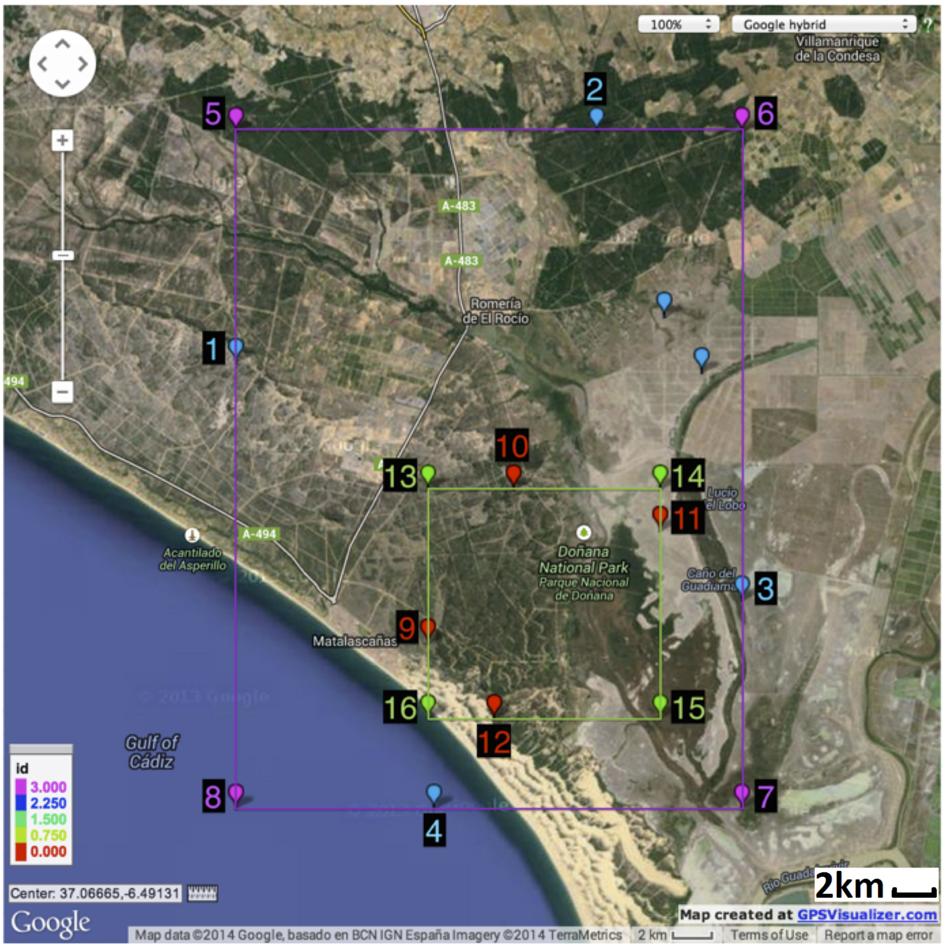


Figure 3.5: Area encompassing the range of the movements of the horses.

3.2.3.1 Calculating the distance between two GPS locations

The distances between two GPS locations were calculated using haversine formula [107], which calculates the great circle distance between two points on the Earth (specified in decimal degrees). The haversine formula is a special case of a more general formula in spherical trigonometry, the law of haversines, relating the sides and angles of spherical triangles. The harversine formula was chosen because it uses "Spherical Trigonometry" instead of a "Law of Cosines" based approach which is based on two-dimensional trigonometry, therefore it offers a nice balance of accuracy over complexity.

This method of calculating the distance between two GPS locations was used in calculating the speed of the horses between locations and in the conversion process of the GPS coordinates into 2D coordinates within the simulator.

3.2.3.2 Conversion of GPS coordinates into coordinates on a 2D plane / grid

Due to the ellipsoid shape and the curvature of the Earth, converting GPS coordinates into coordinates within a plane is not as straightforward as scaling the up or down an area in a 2D plane. When importing the GPS positions into the simulator, it is critical to maintain the same distance proportions between the locations. Due to the way distance is calculated between two locations in a plane, converting a GPS coordinate into a 2D one is not an appropriate approach. Therefore, instead of converting the GPS coordinates, the import of locations into SpeckSim was achieved by converting the distances between these locations.

A reference point was selected, in the bottom-left corner of the aforementioned green square representing the area confining the horses' movements. This reference point is represented in Figure 3.6 by point A. When a new point N needs to have its GPS coordinates converted, the distance between the GPS coordinates of A and of the projections of N on the OX and OY axes are calculated using the haversine formula which takes into account the ellipsoid shape of the Earth. These distances are obtained in kilometres and are converted to the distance-measuring unit used in SpeckSim by being divided by eleven. The offsets of the reference point (0.02 for both the OX and OY axes) are added to the newly obtained distances, in order to obtain the (x, y) coordinates of point N. (see Formula below)

$$N(x,y) = \left(\frac{dist(A(Lon_A, Lat_A), Nx(Lon_A, Lat_N))}{11} + X_A, \frac{dist(A(Lon_A, Lat_A), Ny(Lon_N, Lat_A))}{11} + Y_A \right)$$

Where:

- Lon_A = GPS longitude of point A
- Lat_A = GPS latitude of point A
- X_A = SpeckSim coordinate of point A on the OX axis
- Y_A = SpeckSim coordinate of point A on the OY axis
- Lon_N = GPS longitude of point N
- Lat_N = GPS latitude of point N

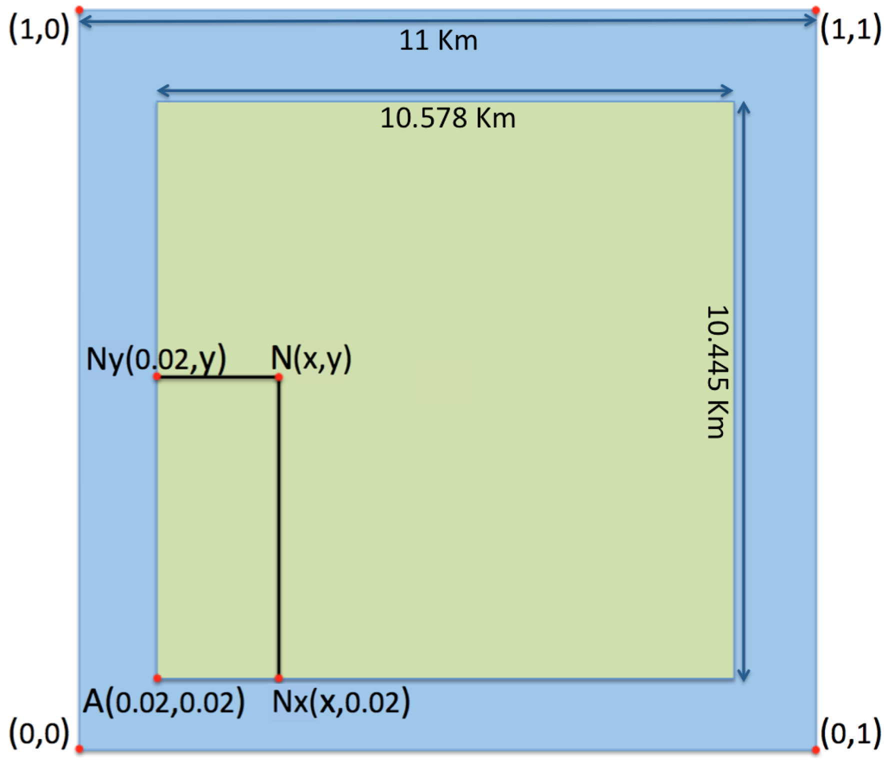


Figure 3.6: Projecting the area containing the horses (the smaller, green square) into the SpeckSim simulation area (the larger blue square).

Figure 3.7 shows as an example the eight main GPS locations that define the area where the horses are located and their matching import in SpeckSim.

3.2.3.3 Setting the radio range

The radio range between a mobile node and a base-station node in the deployment in Doñana is approximately 1000m. This was scaled to a range of 0.09, to match the proportions of the simulation area. The signal propagation model used in SpeckSim degrades the signal strength of a transmission by taking into account a path loss variable, which takes into account the distance that the transmission travels. In order to obtain the desired radio communication range, the radio's transmission power level

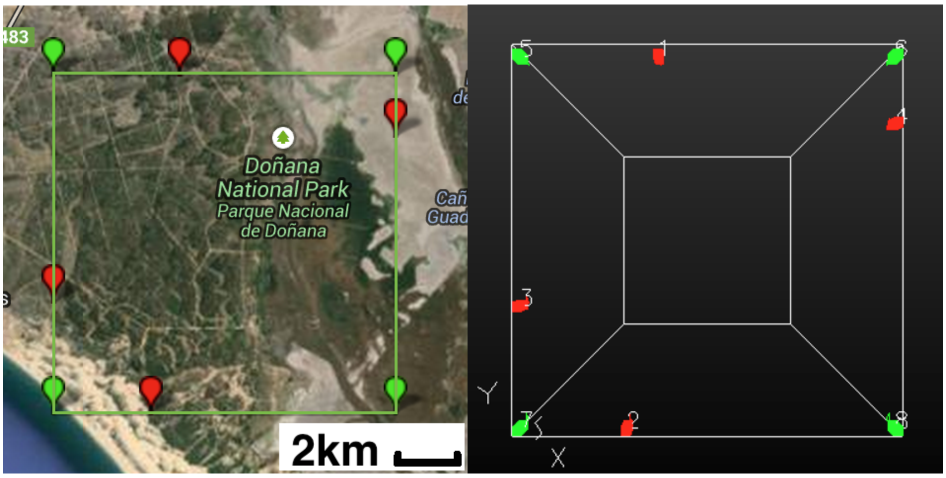


Figure 3.7: Import of GPS locations into SpeckSim.

and the receiver sensitivity need to be set according to the following relationship:

$$TransmissionPowerLevel - PathLoss(distance) > ReceiverSensitivity$$

3.2.4 Initial Simulation Results

This section presents the results of a twelve-month simulation of the scenario of tracking wild horses in the Doñana National Park. The aim of this initial simulation was to estimate the power consumption of the nodes, and thus the deployment lifetime, before the actual deployment was set in motion. At the time of this simulation, no real data was available to input into the simulation models. The models of the hardware components were only based on the figures presented in their datasheets. Considering the availability of mains power for base-stations in the Doñana reserve, the asynchronous protocol was chosen for the deployment. The scenario of tracking and monitoring wild horses was built in the simulator by mapping the area that would confine the horses and by having the mobile nodes moving randomly in this area, at random speeds that can peak up to a horse's galloping speed (40-48km/h or 25-30mph), and that average to a horse's average four-beat walk (6.4km/h or 4.0mph) [108].

When taking the values related to the power consumption of the hardware components from their datasheets and using them for the corresponding simulation models, if these values were expressed as ranges, the higher ones were selected. The reasoning behind this was to obtain the results for a more pessimistic case, but even so, the results looked promising. The mobile nodes lasted over ten months, which is a satisfying deployment length as it covers seasonal variation. A large amount of data was uploaded to base-stations: 85.5% of all the unique packets that would have been col-

Table 3.1: Tracking and Monitoring Horses in the Wild - Simulation

Metric	**Performnace**
Deployment length	Mobile nodes lasted for 310 days or 10.18 months
Average network latency	2 days, 23:27:01
Total number of unique packets collected	184822 (85.5%)
Total number of packets collected	732764
Redundancy 2-way	99.85%
Redundancy 3-way	99.21%
Redundancy 4-way	97.29%

lected during the nodes' lifetimes. Also, the data collection redundancy levels were very high, obtaining a 4-way redundancy of 97.29%. These results were encouraging for taking forward the deployment in reality. The data collection process (amount of data collected, latency, redundancy levels) in such a deployment is highly dependent on the movement patterns of the mobile nodes, thus the results showed in Table 3.1 may not be representative for the horses' real movements in the wild. Nevertheless, the results concerning the mobile nodes' power consumption remain representative, as the horses' movements have little effect on this metric.

3.3 Summary

This chapter introduced the VB-TDMA data upload protocol proposed for the long-term tracking and monitoring of mobile entities in the outdoors, describing its functionality and aspects of its implementation. It presented another protocol, a simple asynchronous one, appropriate only in scenarios with mains-powered base-stations. It further described the environment developed for evaluating the VB-TDMA protocol, consisting of the simulation models corresponding to the hardware components of the Prospeckz-5 platform, along with the node mobility model, and the methods for importing data from real deployments into the SpeckSim simulator. The results of the initial simulation of tracking and monitoring horses, revealed that the asynchronous data upload protocol shows promise of a satisfactory deployment lifetime of over ten months.

The next chapter presents a complete solution (inclusive of hardware) for the long-term tracking and monitoring of mobile entities in reality, for the specific scenario where the mobile entities are wild horses.

Chapter 4

Real-World Deployments

Successful deployment of large scale systems over extended periods of time of several months requires meticulous design and testing which should take into account the environment and local conditions. The focus in real deployments usually lies on the practicality and robustness of the solution. This chapter presents our proposed method for tracking and monitoring wild horses, a representative scenario for the targeted class of applications. The first part of the chapter describes the design of both the mobile and the base-station nodes, their performance in pre-deployment tests, and how this affected the final design. The second part presents a long-term deployment on thirty-two wild horses in a nature reserve using the asynchronous protocol for data collection. The third part looks at a smaller deployment on eight domesticated horses, using the VB-TDMA protocol. In the case of both deployments, an analysis of the sensor data is provided, giving valuable insights into both the individual and the group behaviour of the horses.

4.1 Node Designs and Preliminary Tests

Key issues to be considered in the design of nodes for this class of applications include customising features to the particular mobile entities being tracked, ensuring that all aspects of potential environmental challenges are considered (particularly seasonal extremities inherent in outdoor tracking), and ensuring adequate and reliable power sources for the duration of the deployment. This section will explore the options available and the challenges faced in the design of the mobile nodes that were attached to the horses. It will also describe the design of the base-station nodes and present results from pre-deployment tests.

4.1.1 Mobile / On-body Sensor Nodes

Custom hardware [28] was developed to overcome the challenges related to placing on-body sensor nodes on a rare breed of endangered horses, the Retuerta, in their natural habitat. The Prospeckz-5 platform used in the deployments described in this thesis was designed by Janek Mann at the Centre for Speckled Computing, University of Edinburgh.

As mentioned, the conflicting demands of a long battery life (more than six months), housed in a modestly-sized enclosure (footprint smaller than a ten cm square) and light-weight (less than six hundred grams), have influenced a number of design considerations for the body-worn sensor. The size and weight of the battery (which constituted almost 90% of the interior of the enclosure) was dictated by the welfare of the animals as authorised by the biologists of Doñana.

The sensor node had to be powered mainly off a primary cell, as there were no options for externally recharging the batteries or changing them during the period of study. A dual battery interface enabled operations to be supported by both a Lithium Thionyl Chloride primary cell and Lithium Polymer rechargeable battery. Lithium Thionyl Chloride batteries combine excellent energy density with extended operational temperature range and minimal self-discharge, making them a suitable choice for long-term deployment scenarios. The Lithium Polymer battery can be charged by a solar cell using an efficient energy-harvesting circuit. However, the drawback for this type of battery is that it has a limited range of operating temperatures and capacity. The reasoning was that should the voltage from the lithium polymer battery drop below its operating range, the current will be sourced from the primary cell instead, which was sized to allow operation of the platform for approximately twelve months. However, as will be explained later in this thesis, the Lithium Polymer battery was not used in the final design of the nodes.

The sensor node is based on the custom-designed Prospeckz-5 presented in Section 2.1.3, the latest offering in a platform series first launched in 2004 [5], which combines sensing, processing and wireless networking capabilities. The Prospeckz circuit was potted using hot melt electronic adhesive to reinforce all the wire joints to withstand any shock, and it was stress tested using drop tests from a height of five metres onto a carpeted stone floor.

The chosen enclosure for the on-body nodes is a hard plastic and polycarbonate case with a clear cover - the PN-1320-CMB sourced from the Bud Industries [30].

It provides waterproofing up to IP68 specification level and is rated as a NEMA 4X, 12, 13, *i.e.,* it is weather-resistant with protection against dust, it is watertight and submersible, and it is corrosion resistant. The enclosure was potted with re-enterable silicone compound that provides both vibration protection and waterproofing in accordance with IP68 specifications. The transparent lid of the enclosure was potted with an optically clear silicon encapsulation compound for better fixing the radio patch antenna, while not interfering with the GPS antenna and allowing light to pass through. This enabled the use of the light sensor, the possibility of using a solar cell, and was useful in the initial phases of deployments when the LEDs on the Prospeckz-5 were used for testing the functionality of the nodes.

A fully sealed, enclosed device is better able to cope with the harsh environment, but at a price of compromising the radio performance by relying on a PCB or chip antenna. A u.FL antenna connector offers flexibility in the antenna choices. An enclosure-mounted antenna was used to optimise the radio performance of the nodes. The same PCB was used in the base-stations and mobile data collection systems with larger external antennas, with the performance optimised for each application. Such an approach achieved the required robustness in a cost-effective manner.

The strap, as shown in an early concept design in Figure 4.1, is required to last at least twelve months on the wild horses, as the current animal welfare regulations adopted by the Doñana reserve do not allow capturing these horses more than once a year. The straps are made out of Biothane, a durable weather-proof material. They incorporate the node enclosure and position it against the flat part of the horses' necks, near the top, for optimal GPS reception. Metal pellet weights are included at the bottom, double-stitched into pockets within the belt, to keep the strap the correct way up. The length of the strap is a minimum of 70-80cm, and it is adjustable by 10cm, allowing a close fit on individual horses' necks.

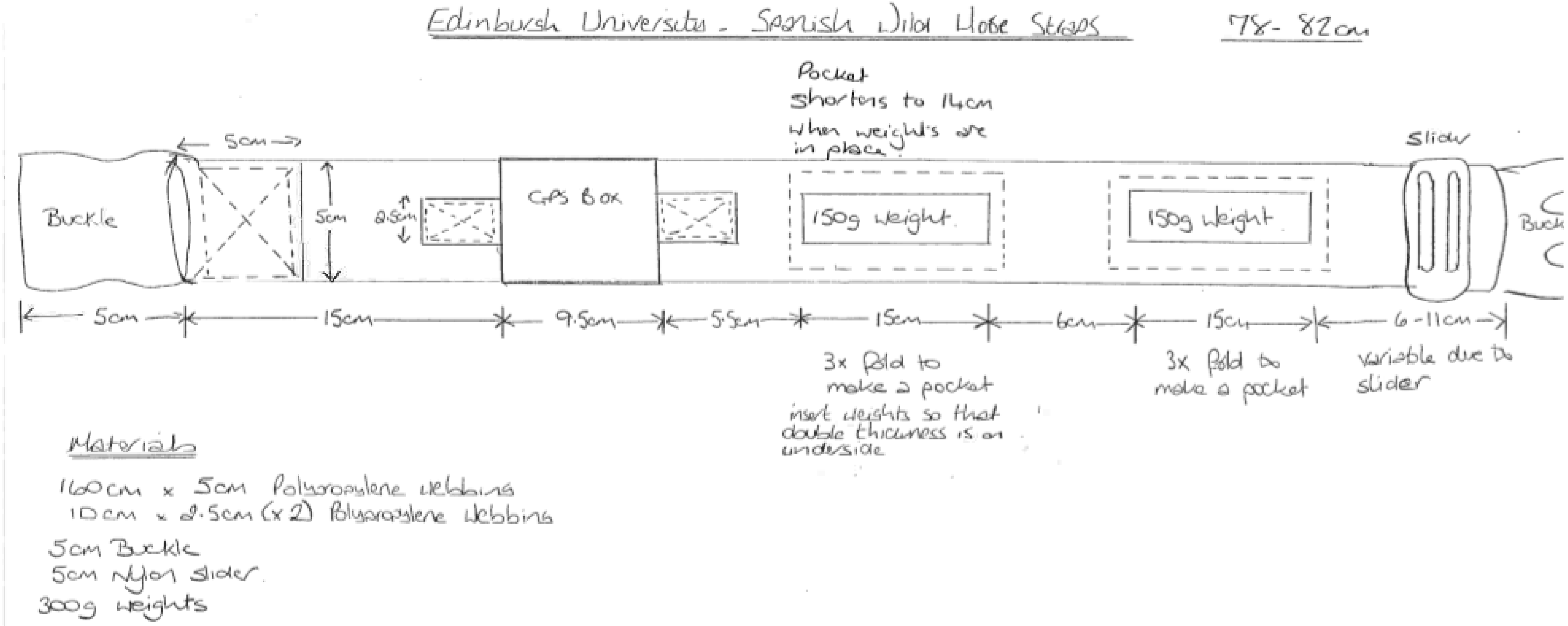


Figure 4.1: Strap design (created by Alison Sangster).

The Prospeckz-5 board is connected to two Lithium Thionyl Chloride batteries (3.6V, 2500mAh) and an enclosure-mounted antenna. The initial design also included a Li-Polymer battery (3.7V, 600mAh) and a solar cell (30x30mm), but due to space constraints inside the enclosure, these were replaced by a second GPS module, the FGPMMOPA6H [109]. The fully assembled on-body node weighs 165 grams without the strap, and approximately 530 grams with the strap.

4.1.2 Base-station Nodes

The base-station node has a larger NEMA 4X rated enclosure containing the Prospeckz-5 board, three Lithium Thionyl Chloride batteries (each 3.6V, 2500mAh), and an external antenna lead. The base-station nodes were initially designed to be mounted on two-metre aluminium poles, mounted into a metal fixing bracket, and secured into the ground by four 60cm-long metal ground anchors. The external antenna, Wifi-Link WLO-2450-08 (8dBi omni), would have been mounted on the top of the aluminium pole to obtain a good radio range for the base-stations. However, due to the existence of towers throughout the Doñana nature reserve which provide 12V power and Ethernet connections, the base-stations were mounted on top of these towers, and consisted of a Prospecks-5 board connected through a serial interface to a Raspberry Pi board and a 12V to 5V DC/DC converter.

Figure 4.2 shows the collars with a mobile base-station on the left-hand side and a static base-station mounted on one of the towers in the Doñana National Park on the right-hand side. Figure 4.3 shows the components of the static base-stations and the enclosures.

4.1.3 Pre-deployment experiments and tests

The Prospeckz-5 platform was subjected to extensive testing over a period of more than four months.

4.1.3.1 GPS and the Sensors

The GPS module, along with the other in-built sensors on the Prospeckz-5 platform, were tested individually to assure their correct functionality. There is a trade-off between the positional accuracy and the time for which the GPS module is on, which directly translates to power consumption. We managed to significantly improve the

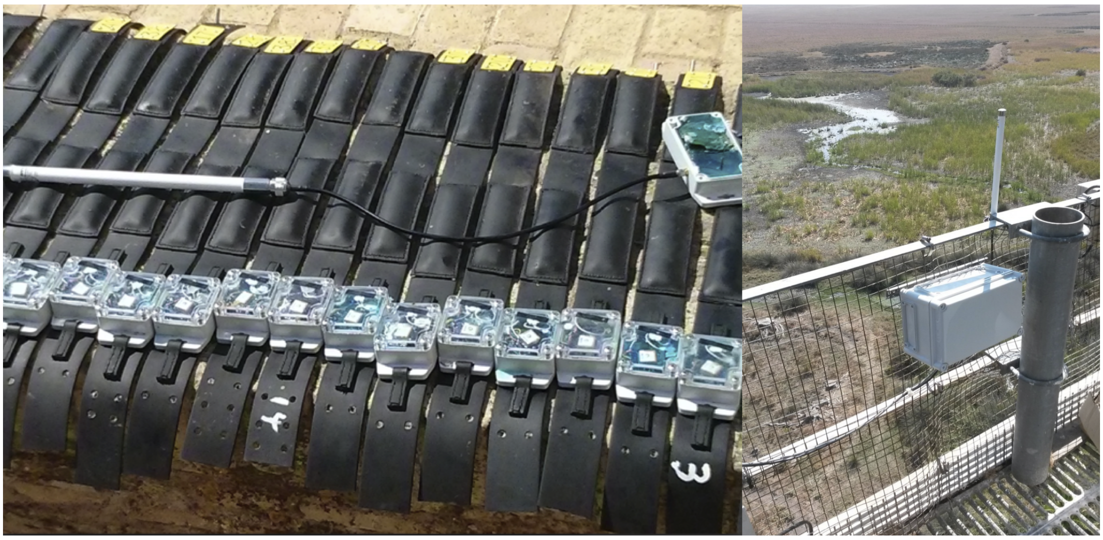


Figure 4.2: An arrangement of mobile nodes ready for deployment with a base-station and antenna (left). Base-station on top of a tower in Doñana National Park (right).

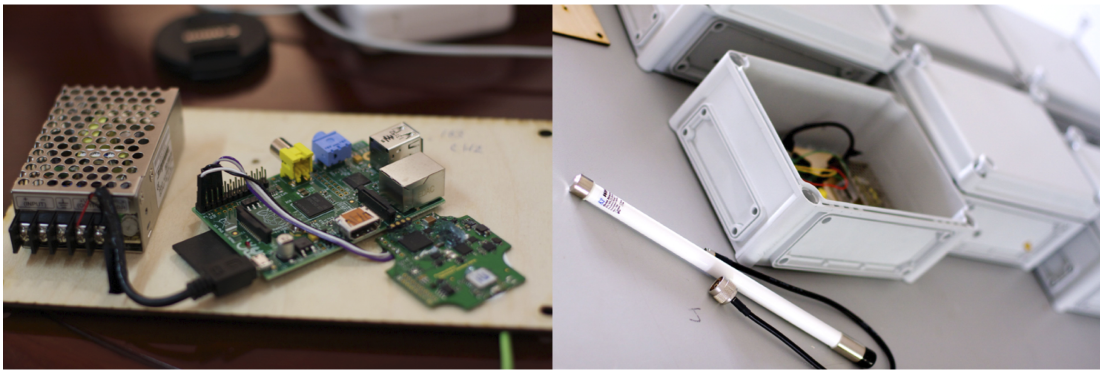


Figure 4.3: Base-station components: power supply, Raspberry Pi and Prospeckz-5 (left). Base-station boxes and omnidirectional antenna (right).

accuracy by keeping the GPS module awake three seconds longer after obtaining a fix, and nineteen seconds longer after every five twenty-minute intervals. This allows it to gather more information about the satellites, by updating the ephemeris data (used to calculate the position of each satellite in orbit) and the almanac (information about the time and status of the entire satellite constellation), and provide a more accurate position at the end of the extra time (see Table 4.1 and Figure 4.4). The accuracy is expressed in terms of Circular Error Probability (CEP). CEP50 is defined as the radius of a circle centred on the true value that contains 50% of the actual GPS measurements. Thus, a receiver with a one-metre CEP accuracy will be within one metre of the true measurement 50% of the time. It can be observed in Table 4.1 that the CEP improvement lowered CEP50 to 9.56 metres and CEP98 to 58.14 metres. Figure 4.4 illustrates

Table 4.1: Comparison of GPS Accuracy Measurements

CEP	Accuracy	
	Initial Accuracy	**Improved Accuracy**
CEP50	13.38	9.56
CEP90	83.45	58.14

this graphically, showing the GPS locations of the collars deployed statically in a small area in the Doñana Park. The right-hand side of this figure shows the locations of the collars with the improved accuracy, which did not generate obvious outliers such as the ones displayed on the left-hand side.

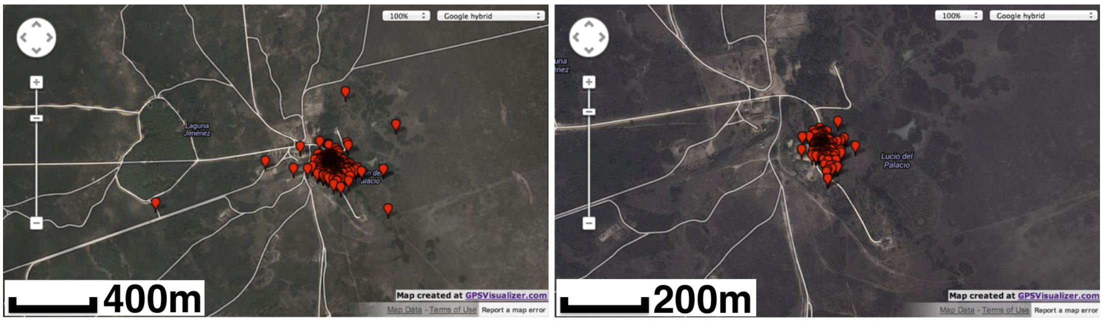


Figure 4.4: Location accuracy improvement: initial accuracy (left), improved accuracy (right).

4.1.3.2 Solar Cells and Battery Charging Circuit

When the use of solar cells was being considered, three sizes were tested: small (12x40mm), medium (30x30mm) and large (50x60mm). The experiments were performed at different times of the day using a 10kΩ and a 1kΩ load resistor. Table 4.2 shows that during peak sun exposure the cell operates at close to its open circuit voltage with the 10kΩ resistor, and therefore does not realise its optimal power. In contrast, with a 1kΩ load, a higher power output is produced. Taking these results into consideration and the fit of the solar cells in the lid of the enclosure, the medium-sized one (30mmx30mm) was deemed most appropriate. The battery charging circuit was tested in direct sunlight to produce an 8-10mA charging current, and 1mA charging current in the shade.

Table 4.2: Solar Cells Measurements

Midday/Afternoon (Sunny) 15:00 - 15:30, July 1st 2013, Villamartin, Spain				
Solar Cell	**With 10kΩ**		**With 1kΩ**	
Large	6.20 - 6.30V	3.84 - 3.97mW	5.50V	30.25mW
Medium	3.21 - 3.27V	1.03 - 1.07mW	3.05V	9.3mW
Small	2.36 - 2.37V	557 - 562μW	0.83V	688.9μW
Evening (Sunset) 21:05 - 21:20, July 1st 2013, Villamartin, Spain				
Solar Cell	**With 10kΩ**		**With 1kΩ**	
Large	5.50 - 5.95V	3.02 - 3.54mW	0.82V	672.4μW
Medium	2.60 - 3.04V	676 - 924μW	0.39V	152.1μW
Small	1.40 - 1.83V	196 - 335μW	0.13V	16.9μW

Table 4.3: Enclosure-Mounted Antennas Tests

Position	Ethertronics (dbm)	Taoglas Limited (dbm)
South	-36.50	-40.00
North	-33.50	-40.00
East	-35.67	-45.50
West	-45.50	-47.50
UpTowardsScope, West Down	-35.00	-34.00
UpTowardsScope, East Down	-35.50	-39.50
DownTowardsScope, West Down	-37.50	-44.50
DownTowardsScope, East Down	-35.00	-40.50

4.1.3.3 Enclosure Mounted Antenna

Nine antennas were considered and the "Prestta" (ethertronics.com) was chosen based on its size (44x7.7x0.85mm), shape and bench test performance. Table 4.3 summarises the bench tests performed for the first two options for the enclosure-mounted antennas. The tests were conducted with the antennas mounted on the inside of the lid of the enclosure, as is the case during deployment. A radio test application was developed for the Prospeckz-5 platform emitting a constant carrier from the antenna port. The strength of the field from the antenna was measured using an HP spectrum analyser in bench conditions. Two possible placements of the antenna next to the PCB were evaluated, as the ground plane of the PCB in proximity with the antenna impacts its performance.

4.1.3.4 Antenna Radio Range

Two external antennas, Wifi-Link WLO-2450-08 (8dBi omni) and WLO-2450-12 (12dBi omni), were considered for the base-stations, and were tested when mounted at a height of two metres above ground. The WLO-2450-08 demonstrated better performance with a communication range of around 500m, which was likely due to the narrow horizontal radiation pattern of the higher gain antenna. The following two cases were tested: the on-body node as a transmitter, and the base-station node acknowledging the received packets, and the other way around. The Packet Reception Rate (PRR) was measured for different node distances, calculated based on the GPS location of the mobile on-body node with respect to the location of the fixed base-station. Tests were also conducted using multiple on-body nodes communicating to the same base-station.

The WLO-2450-08 was further tested against a unidirectional antenna in the Doñana Park, in the environment of the deployment. The tests were conducted on a flat surface that included dense tree-coverage, with the base-station antenna maintained at a height of two metres. The results show that in the case of the omnidirectional antenna the maximum range was approximately 558m and for the unidirectional one it was approximately 655m. The area where the antenna range tests were performed can be seen in Figure 4.5. When the base-stations with the WLO-2450-08 omnidirectional antenna were mounted on the towers, the communication range increased to approximately one kilometre.

4.1.4 Node Design Tradeoffs and Experiences

As stated previously, the Prospeckz-5 design was driven by several external constraints, primarily the requirement for an enclosure small enough to suitable for attachment to the horses and the need for a long-term battery lifetime (ideally over twelve months). The enclosure was one of the first considerations during the design of the Prospeckz-5. Due to the relatively small number of devices which were going to be manufactured, it was decided that it was not practical to develop a custom enclosure for the project, thus a number of off-the-shelf enclosures were considered. The casing had to be watertight, shock-proof, lightweight, and with mounting features to allow attachment to a strap around the horses' necks. A transparent window was desired for the casing to enable the use of a solar cell inside. A number of cases were considered, and the PN-13XX range from Bud Industries was chosen for its NEMA-4X rating and the transparent polycarbonate top lid. The PN-1320-CMB, in particular, was the smallest that could

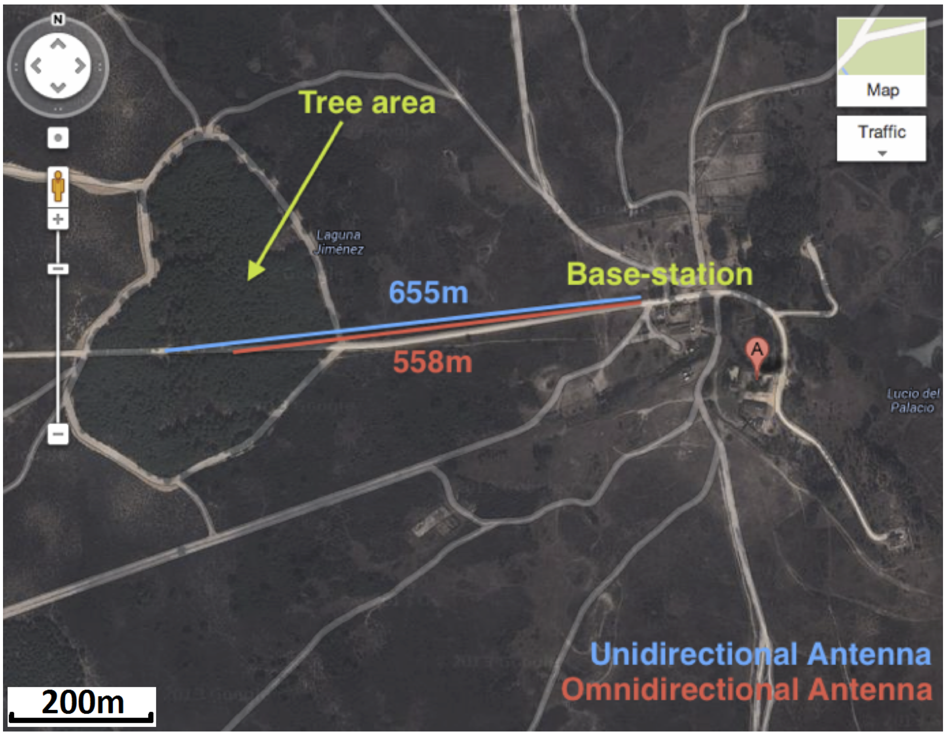


Figure 4.5: Antenna range tests in the Doñana National Park.

accommodate the chosen batteries. The available space inside this enclosure defined the shape of the custom-designed PCB for the Prospeckz-5.

A number of battery topologies were considered, including rechargeable ones using solar power. It was deemed too risky to be dependent on solar power or rechargeable batteries, as solar power can be disrupted by mud coating the window to the solar cell, and rechargeable batteries have limited performance in extreme temperature conditions as can be encountered in the environment the Prospeckz was going to be placed in. It was decided to use a pair of AA Lithium Thionyl Chloride batteries [110] as primary cells, as these possess the highest commonly available energy density (approximately 160mAh/g) as well as being rated for extended temperature ranges (minimum -55°C, maximum 85°C). They are particularly well suited to handle the long-term low average current needs of the Prospeckz-5.

The design criteria for the microcontroller (MCU) at the heart of the Prospeckz-5 were ultra-low power operation together with a reasonable amount of RAM to allow for large buffers to minimize access to the serial flash memory, which is relatively energy-intensive during each access due to the need for erasing parts of the flash. The EFM32GG990 was chosen due to the high performance of the Cortex-M3 core [111], which minimises the active time of the processor (during which it consumes considerable current - 200μA in the Run Mode state EM0) and the very low current consumed during idle times (1.2μA in Deep Sleep Mode state EM2). Another trade-off was the size of the MCU package in relation to the amount of RAM available, and the chosen MCU is one of the smallest with a large amount of RAM (1024KB).

A GPS module with an integrated antenna was desired to avoid the need for feeding wires outside the enclosure (potentially affecting the water-proofing), and to have minimal footprint, leaving space for the rest of the circuitry on the small PCB. The UC430 module chosen provided both of these features, and claimed support for ultra-low-power modes that should have allowed the module to operate at a very low average power (most of its time would have been spent in a Hibernate state where the module achieves 125μA typical average current drain). Unfortunately it turned out that this module, due to both the mounting on a horse as well as the limited ground plane available on the space-constrained PCB, did not operate with sufficient antenna performance to take advantage of this energy-saving feature. Therefore, the deployment in the wild used a different module, the FGPMMOPA6H, with a larger ceramic antenna, which was mounted on top of the Prospeckz-5 PCB. This module does not have the ultra-low-power mode, but offered a considerably superior antenna performance along with a lower current consumption when active.

Based on our Research Centre's experience with ultra-low-power radio operation, a 2.4GHz transceiver (NRF24L01+) was chosen with the option of using a relatively high signal rate of 2Mbps. In most designs it is customary to choose a transceiver with much lower signal rate but with a higher receiver sensitivity, which would enable a signal to be picked up under bad propagation conditions. However, in the case of Prospeckz-5, its design favoured an extremely low power per bit transmitted, making it cheaper in terms of the power consumption to send a packet, thus allowing sending "probe" packets more frequently in return for better network responsiveness. This radio was augmented with an RFX2401C amplifier IC, which combines both a PA (power amplifier) for amplifying the transmitted signals as well as a LNA (low noise amplifier) for amplifying the received signals before processing by the radio receiver. The combination of the NRF24L01+ with the RFX2401C can offer an end-to-end gain of up to 24dBi.

An internally-mounted antenna was chosen for the 2.4GHz radio, to keep the casing as water-tight as possible. An internal antenna connector was preferable to a PCB-antenna or other PCB-mounted antenna to retain flexibility in the choice of antennas, for testing and choosing one based on the performance inside the enclosure. Therefore, a selection of antennas suitable for mounting inside the enclosure was evaluated (Section 4.1.3.3).

The power supply of the Prospeckz-5 had to be optimised both for extracting the maximum amount of energy out of the batteries as well as minimising the idle current drain. The batteries supply a voltage between 3-4.2V while the components on the system themselves operate at 1.8-2.2V. Using a purely linear voltage regulation strategy (using LDOs for example), nearly half the energy in the batteries would have been wasted. Typical switching DC-DC regulators, which can regulate the voltage much more efficiently, have relatively large idle currents. The Prospeckz-5 components are carefully chosen to offer an overall idle current in the order of 10s of μA, while typical switching regulators can have idle currents in the 100s of μA. We chose the LTC3103 switching regulator, which has an idle current of only 1.8μA due to its burst mode of operation. As a result, the power from this regulator is relatively noisy, thus additional low-noise LDO regulators are used for the GPS module and the flash to ensure good performance. For the solar charging feature (which was not used in the deployments described in this thesis) the ability to operate from a very low solar flux, maximum power point tracking to extract the optimal energy from the cell, as well as battery charging features were desired. The initial design of the Prospeckz-5 chose the LTC3105, which promises these features but did not operate to our satisfaction in our application. The BQ25504 solar charging IC from TI, which fulfilled these requirements, was chosen for the final design.

4.2 Long-term wildlife deployment

The Retuerta, one of the oldest horse breeds in Europe, roams wild in the Doñana National Park, Andalusia, Spain. Thirty-two of these horses were marked with wireless sensors to gather spatio-temporal data. This section describes the tracking and monitoring of these wild horses wearing collars with sensors that operate in a harsh and challenging environment. Analysis of the data has revealed rare insights into the horses' social behaviour, such as group dynamics (group sizes and memberships), dispersal and home ranges which are of interest to both animal ethologists and practitioners managing the ecology of their wild habitats. As discussed, a number of choices had to be made to address technical challenges such as the design of the sensor platform, wireless data collection and battery lifetime issues for an extended deployment. The experiences point to the virtue of simplicity in the design of wireless sensor networks to support core functionalities for achieving good average case performances.

4.2.1 Scenario

4.2.1.1 Justification

The high-resolution data of the location and activity of animals can be used for movement ecology analyses and conservation studies that lead to gaining better understanding of environmental and biological factors that relate to animals' movement patterns and behaviour.

Animal tracking can expose the effects of the monitored animals over the vegetation, and the other way around, how the animals move in relation to the quality of vegetation, and it can also reveal information about group relations. There is also interest in monitoring and analysing direct contacts between individuals. Since infectious diseases represent a threat for the biodiversity conservation and they spread through direct contact between individuals, such a study can provide valuable information about the way these diseases spread. "Nearly 75% of emerging infectious diseases had their origin as zoonoses (infectious diseases that can be transmitted between species). Vectors such as mosquitoes can transmit some zoonoses, but many other require direct contact or proximity of individuals for its transmission. This is the case of avian flu, tuberculosis, Newcastle disease or measles. Although the pathogens may remain viable in the environment for a limited amount of time, direct contact of individuals greatly facilitates the transmission." [112]

4.2.1.2 Motivation (the technological challenge)

The capturing of any group of wild animals is rife with complications in terms of logistical problems and welfare considerations. Having data stored locally on a device attached to an animal, and recapturing it as required in order to retrieve the data, is not a feasible option. The most practical solution is for animals to wear devices capable of uploading the data wirelessly to allocated collection points. These collection points can be strategically placed to cover large areas of land while focussing on places most frequented by animals, such as water bodies, or areas where the animals forage.

However, wireless communication and gathering GPS location information both entail high power consumption for mobile devices. As such, there is a clear trade-off between the potential runtime of any deployment and the physical dimensions of the devices that are attached to the animals. To account for seasonal variations, these deployments ideally need to run for extended periods of time. This poses significant technological challenges: in terms of hardware, it has to be robust, compact, resistant to a wide range of extreme weather conditions, and at the same time to offer good performance with low energy consumption; and in terms of algorithms, the design needs to be robust and power efficient, having the ability to recover from any state and maximising the potential of the hardware.

4.2.1.3 Scenario description

The aim of this scenario is to track and monitor the behaviour of thirty-two wild horses, over a period that is long enough to capture the seasonal variation. Gathering information on the location of the horses and monitoring their activity and head orientation is done using a GPS module and an accelerometer. Biologists require a tracking (location) granularity of once every 20-30 minutes, for the duration of the deployment. The data collected by the mobile nodes attached to the horses is wirelessly uploaded to base-stations deployed throughout the nature reserve and then forwarded to a central device with network connectivity which uploads it to a server. However, due to the availability of towers throughout the reserve that provide both power and network connectivity, the base-stations are already connected to a private local network via Ethernet. Thus, the data can be accessed remotely straight from the base-stations' SD cards, by setting up a VPN tunnel to this network.

4.2.2 Results

This section presents the outcome of the deployment in the Doñana National Park on thirty-two wild horses, using the asynchronous data upload protocol described in Section 3.1.1.

4.2.2.1 Deployment Aspects

The location information of the horses can be analysed to detect absence of movement over prolonged periods of time, which can indicate that a horse might be injured, sick or dead. The absence of movement can also be flagged by looking at the maximum activity levels of the horses, calculated using the accelerometer data. Also, signs of illness can be deduced from the head angle of the horses (head-down percentage over time). One of the marked horses was found dead after approximately seven months in the deployment. When it was found, its body was almost fully decomposed. The

Figure 4.6: An arrangement of mobile nodes with a base-station and antenna ready for deployment (left). Wild horses being marked with sensors (right).

biologists determined that its death was related to giving birth. Although we had a script ready for detecting such cases (the absence of movement and activity), the horse unfortunately died out of reach of any base-station. Thus the script did not have access to the data showing the horse's inactivity until after it was found and the collar retrieved. Running the script on the data revealed that the horse died only approximately three weeks before its remains were found. The biologists argued that the high decomposition rate is typical in the Doñana Park due to its richness in bacteria and insects.

With the current geographic distribution of the base-station nodes (see left-hand side of Figure 4.8), it took on average six days for a packet to be uploaded to one of the base-stations. This is not surprising given the unpredictable and general slow nature of the movement of horses, and also considering that the deployment of the base-stations was staggered over weeks after the horses were marked with the mobile nodes. The mobile node has an eight-month buffer for outgoing radio packets. The data is also written to the flash memory as a backup, to be accessed upon recovery of the nodes at the end of the deployment. The flash has the capacity of storing multiple years of data gathered by a mobile node, and the data stored on the flash is not lost when the batteries run out.

The horses were recaptured after approximately eleven-twelve months and the collars were removed. The SD cards from all of the base-stations were collected, except one (that of base-station B2 from the left-hand side of Figure 4.8), and it was arranged for all the collars to be placed near this base-station to upload the remaining data.

There were two unexpected setbacks in this deployment that could have been avoided with better planning. Our experiences highlight the need for close on-site monitoring and more rigorous testing prior to deployment.

Given the geographical constraints between our research centre in Scotland and the park in Spain, there was a two-hop communication system in place. At the end of the deployment, when the collars were removed from the horses, there was miscommunication regarding to which base-station these would upload the remaining packets, and they were left within range of the wrong base-station (which had no SD card). Although we realised that no additional packets were being received, it was assumed that this was due to the loss of power on the mobile nodes sooner than we predicted. Once we identified that they were at the wrong base-station, the collars were shipped back to our research centre in Scotland. Unfortunately, the additional power consumption implicit in the shipping process (with GPS in Acquisition state for approximately 9.6 minutes every twenty minutes) lead to the collars' being fully discharged by the time we received them, and were unable to upload packets over the radio.

As previously discussed, as part of the planning process it had been decided to incorporate flash chips into the design, which would provide a fallback in the event of radio failure or unforeseen power losses. However, the second setback faced in the deployment was that we were unable to recover the remaining data packets from the flash chips. Although some testing was carried out prior to using them in the deployment, with hindsight the testing phase on this particular component should have been more rigorous, to ensure that they would be robust enough to operate reliably.

In spite of these setbacks, we were able to gather a significant amount of data on the horses' movements via the wireless upload for a period of approximately nine months, which gave us a good range of seasonal variations. The full potential of the battery life on the devices was not exploited in this particular deployment. However, based on the base-stations' timestamps, the lifetime for the mobile nodes was slightly over ten months, with the final data packets collected a few weeks before the actual upload date.

Due to the three-way redundancy, packets are duplicated on different channels. A visual representation can be seen in the top part of Figure 4.7. The low redundancy percentages are explained by the unequal volume of packets received by each base-station, with most packets being received by base-station B2 on one channel.

The bottom part of Figure 4.7 shows that the base-stations that received the most packets are B1, B2 and B6, suggesting that these base-stations are positioned in locations frequently visited by the horses. The map presented on the left-hand side of Figure 4.8 shows the locations of the base-stations and their assigned channels, and

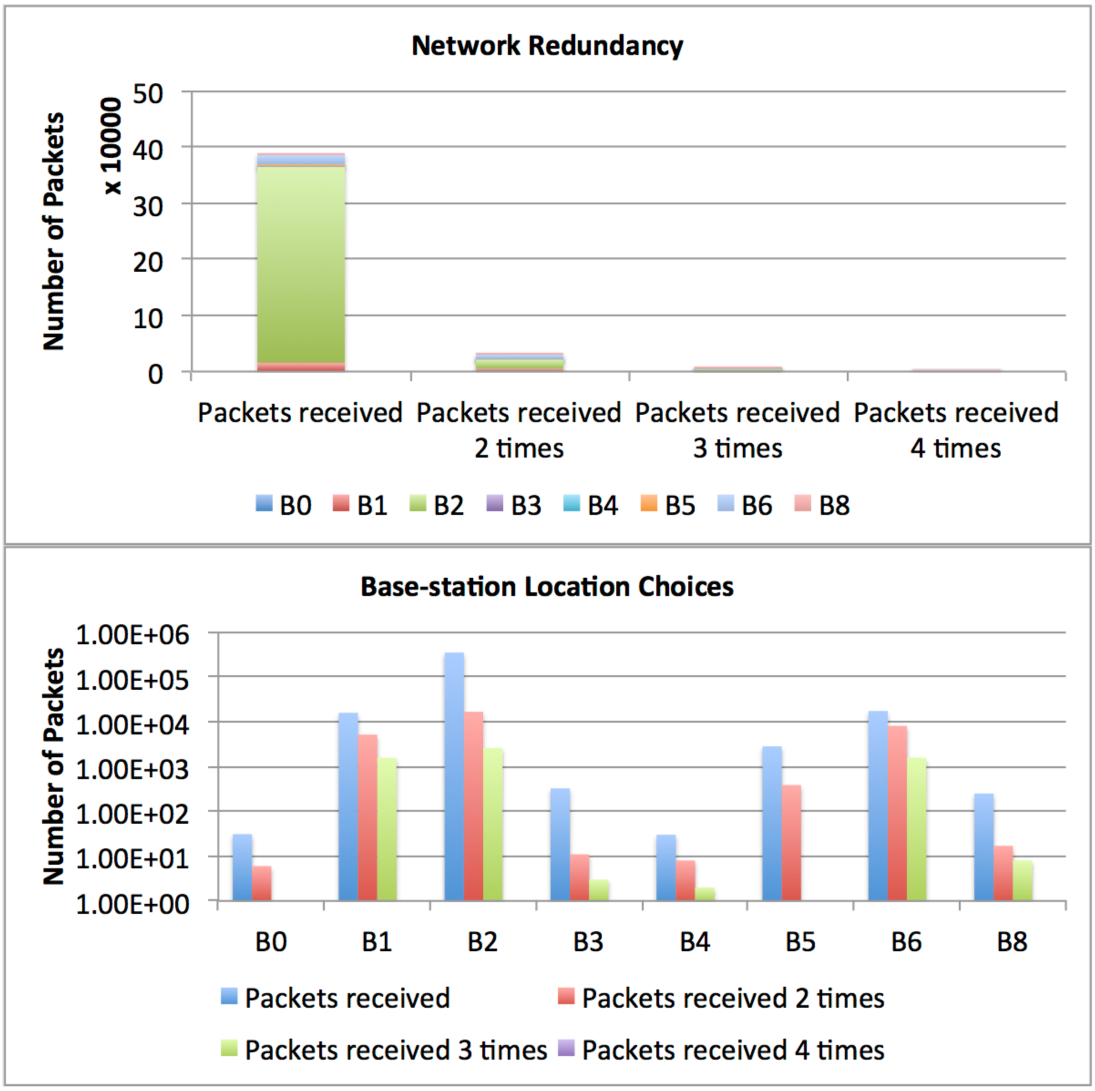


Figure 4.7: Network redundancy (top) and base-station locations choices (bottom).

highlights the area determined by the three base-stations with higher upload traffic, which is assumed to be more frequented by the horses in comparison to the other base-stations' locations. This fact is validated in the right-hand side of Figure 4.8, with green and red colours indicating low and high concentrations of GPS locations, respectively.

The hourly upload pattern of a base-station can peak to 1500 packets per hour, and the information of a base-station's hourly upload pattern can be used to:

- Validate the GPS positions gathered by the mobile nodes, should the timestamp within the packet be close enough to the upload time. This implies that the GPS position should be within radio range of the base-station's position.

- Determine a rough position estimate of a node in case it did not manage to get a

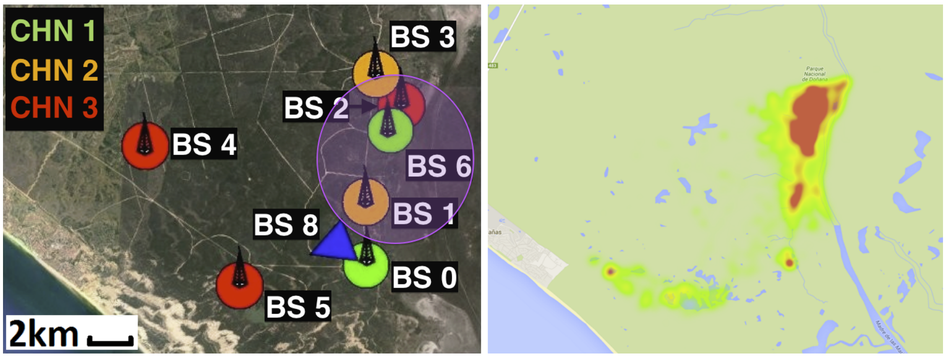


Figure 4.8: Base-station locations (left); heatmap of the horses' GPS locations (right).

GPS fix, and the GPS timed-out for the corresponding twenty-minute time slot.

4.2.2.2 Horse Behaviour

The information that can be extracted from the sensor data is valuable to biologists and ecologists, and validates our solution's ability to monitor the behaviour of the animals. The intention behind the data analysis results presented in this section is to offer an overview of the type of information that can be deduced from the sensor data about the marked horses. The analysis of sensor data also explores the extent to which an asynchronous protocol can be used to infer group behaviour of horses over an extended period of time.

We have chosen to present results from the first two months of the deployment in parallel with the results from the entire deployment. There were two reasons to also look at the first two months. Firstly, Figure 4.9 shows that the first two months have the highest peak in the data collection, and a richer set of data can provide better group behaviour results. Secondly, the idea behind the wireless uploading of the data is to offer access to it during the deployment, so that the horses' status can be checked when needed. Analysing an initial slice of data shows what information was available to the people monitoring the animals after the first two months.

The data for each horse, gathered every twenty minutes, includes the following information collected during the previous interval: the maximum activity based on an algorithm proposed by Mann et al. [113], the percentage of time when the head was down, and the GPS location with its corresponding time stamp. The algorithm used for calculating the activity levels was calibrated against energy expenditure in humans. This calibration has not yet been performed for horses, thus the algorithm is used as a general indicator of activity. It works by determining an estimate of energy con-

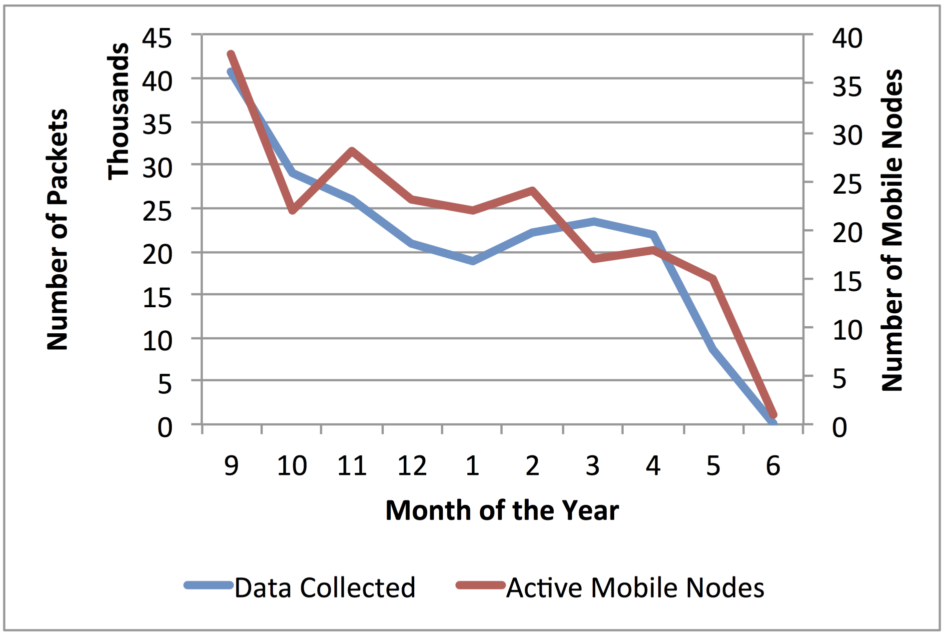


Figure 4.9: Amount of data collected per month (left-hand side axis) and the number of active mobile nodes per month (right-hand side axis).

sumption from calculating the maximum differences between successive accelerometer measurements. This removes the effect of gravity and is correlated to the energy expended when effectuating a movement of the core mass of the body. For determining the head orientation, the angle of the collar with respect to gravity is calculated for determining whether the horse's head is up or down in relation to the ground. Thresholds for two of the axes of the accelerometer were determined by iterative refinement based on observations made on domestic horses carrying the collars. The thresholds were chosen to be tolerant of the collar's orientation, to account for its slipping on the horse's neck.

Figure 4.10 shows the average activity levels for all horses over the diurnal cycle. The pattern validates the known behaviour that horses rest and sleep during the night and are more active during the day. There is a visible resemblance between the activity data over two months and over nine months, confirmed by a strong positive Pearson's r [114] correlation of 0.98. This type of correlation is used throughout the thesis. This shows that the activity of the horses from the initial two months is representative for the entire deployment duration of nine months.

Figure 4.11 shows the hourly pattern of the percentage of time the horses have their heads down over the same periods: two months and nine months. The two-month slice of data seems to be slightly less representative for the head down percentage compared

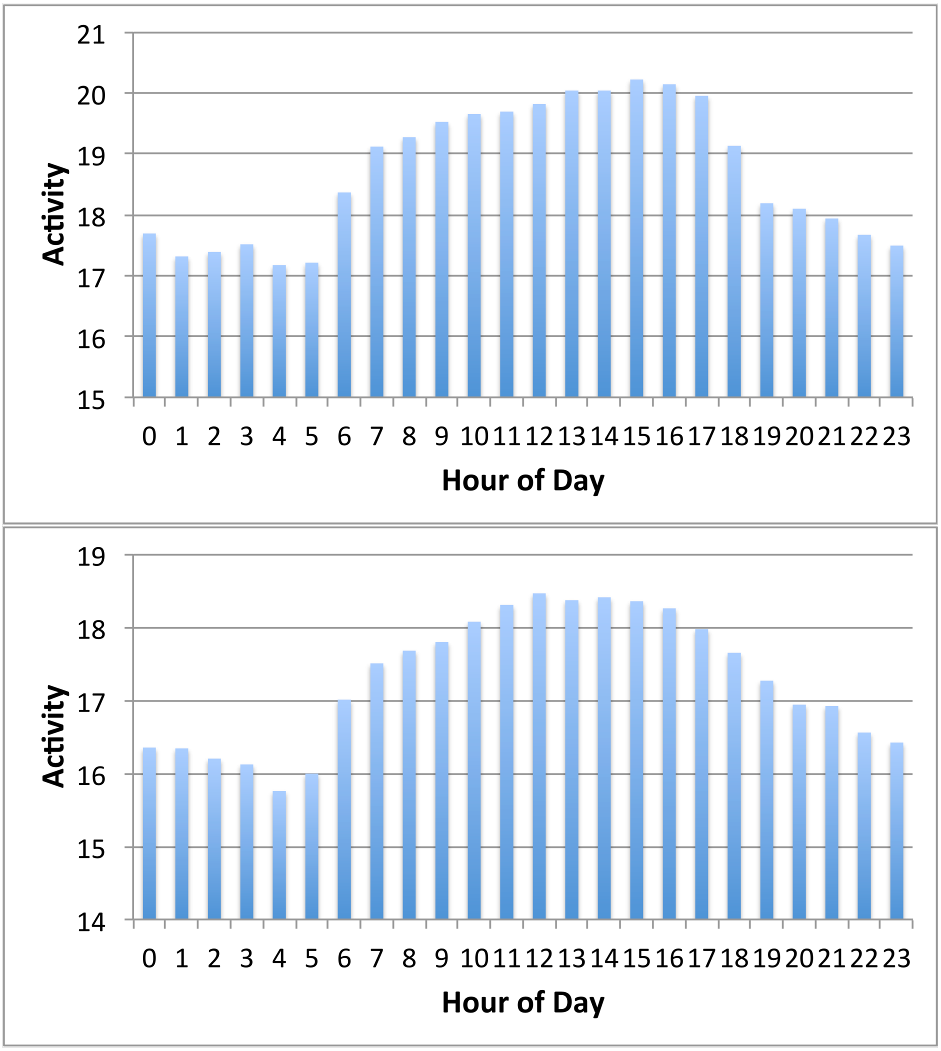


Figure 4.10: Hourly average activity pattern for all the horses over two months (top) and over nine months (bottom).

to the activity, but it is still strongly correlated to the nine-month set of data, with a positive correlation of 0.83. The head being down can be the result of the horse feeding or resting lying down. Horses are able to sleep standing up thanks to specially adapted legs with "stay apparatus" which enables them to relax their muscles and sleep without collapsing, but they need to spend between few minutes to several hours every day lying down [115]. This graph does not disambiguate between the horse grazing and sleeping when the head is mainly down.

The series of graphs in Figures 4.13 and 4.17 illustrate the group behaviour and group dynamics of the horses under study. A horse is considered to belong to a group if it is within 100 metres of any other horse during an hourly interval. Figure 4.12 guided the choice of the distance of 100m based on the sensitivity analysis. The figure

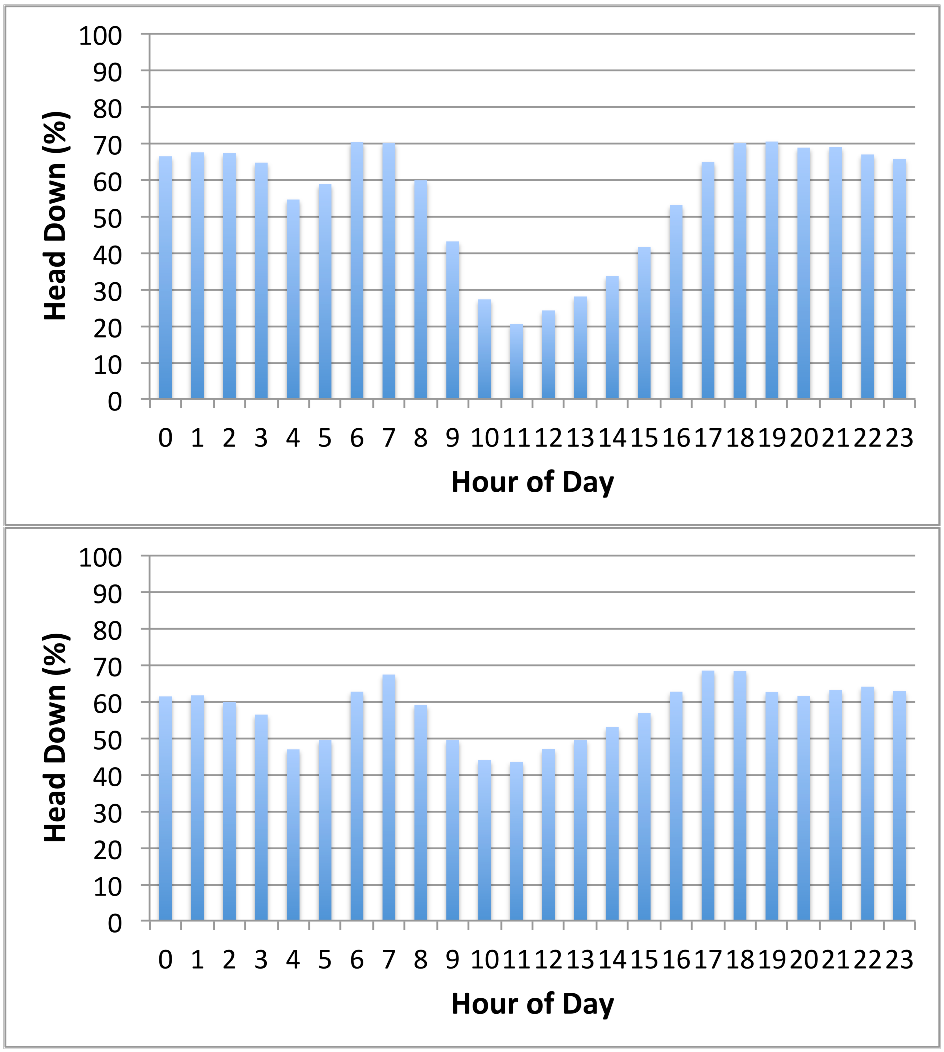


Figure 4.11: Hourly average percentage of head-down time for all the horses over two months (top) and over nine months (bottom).

shows that from D100 to D150 there are almost no changes in the groups' structure (D100 tracks D150 almost perfectly) in the case of both the two-month and nine-month datasets. The 100-metre distance was further validated by checking the composition of the groups that persisted over time, and it was determined that these were mostly composed of one male and several females, confirming the observed "harem" group behaviour of horses in the wild [116, 117]. Figure 4.13 shows the percentage of time that a horse belongs to any group. The majority of the horses belonged to a group at least 80% of the time, which is in keeping with their natural herd instinct as they are social animals, rarely solitary by nature [116].

Figure 4.14 shows the number of instances of group sizes which formed during the period under study. In the top part of this figure it is shown that groups of certain sizes

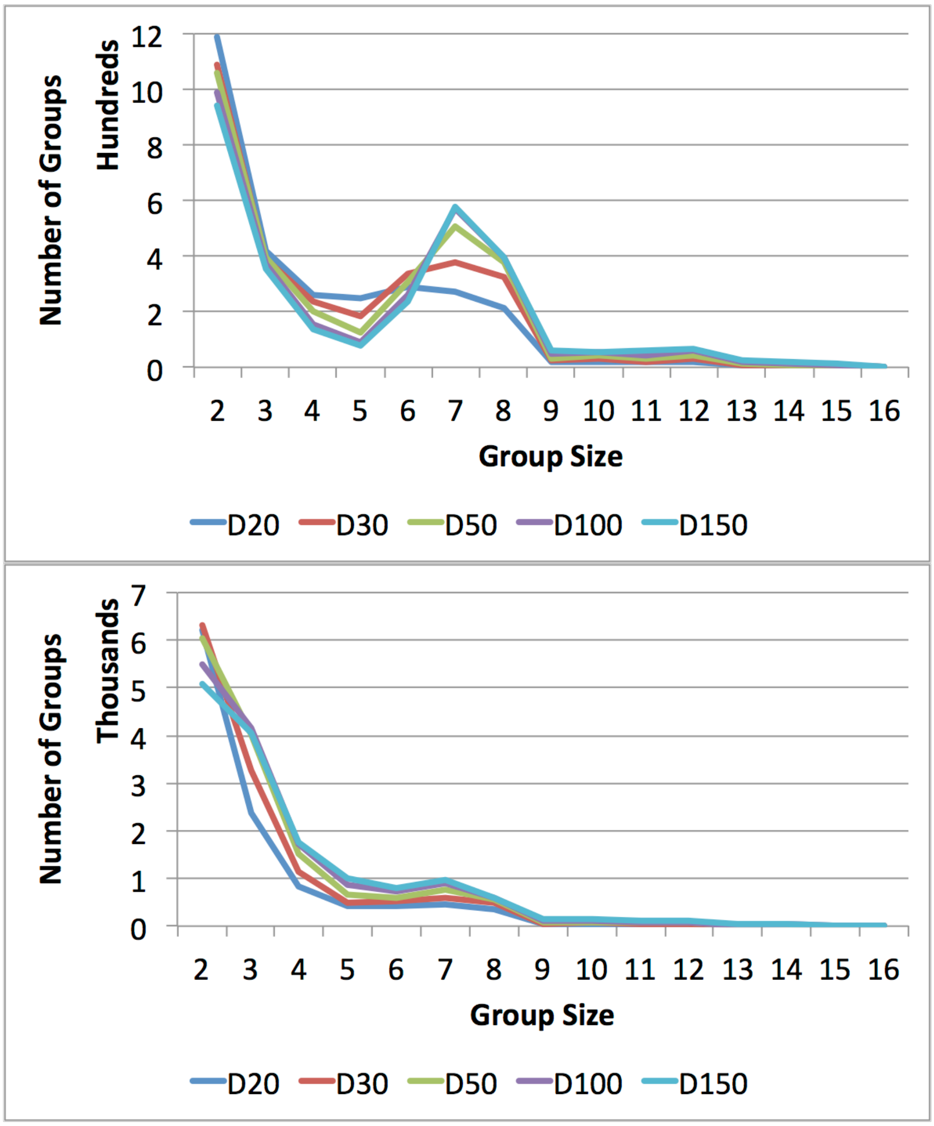


Figure 4.12: Distance sensitivity analysis: two-month set of data (top), nine-month set of data (bottom).

were more common than others: for example, instances of groups of seven horses were the second most popular after groups of two horses. The distribution of the group sizes changes from the two-month dataset to the nine-month dataset. Although there may be changes in the horses' groupings after the initial two months, the differences shown in Figure 4.14 are also due to the lower density of data over the nine months compared to the one over two months (as highlighted by Figure 4.9). A lower data density naturally leads to an increase in the number of instances of smaller groups, which is precisely what is happening in this case as can be observed in Figure 4.14.

Figure 4.15 shows a snapshot of the horses in the different groups and their lo-

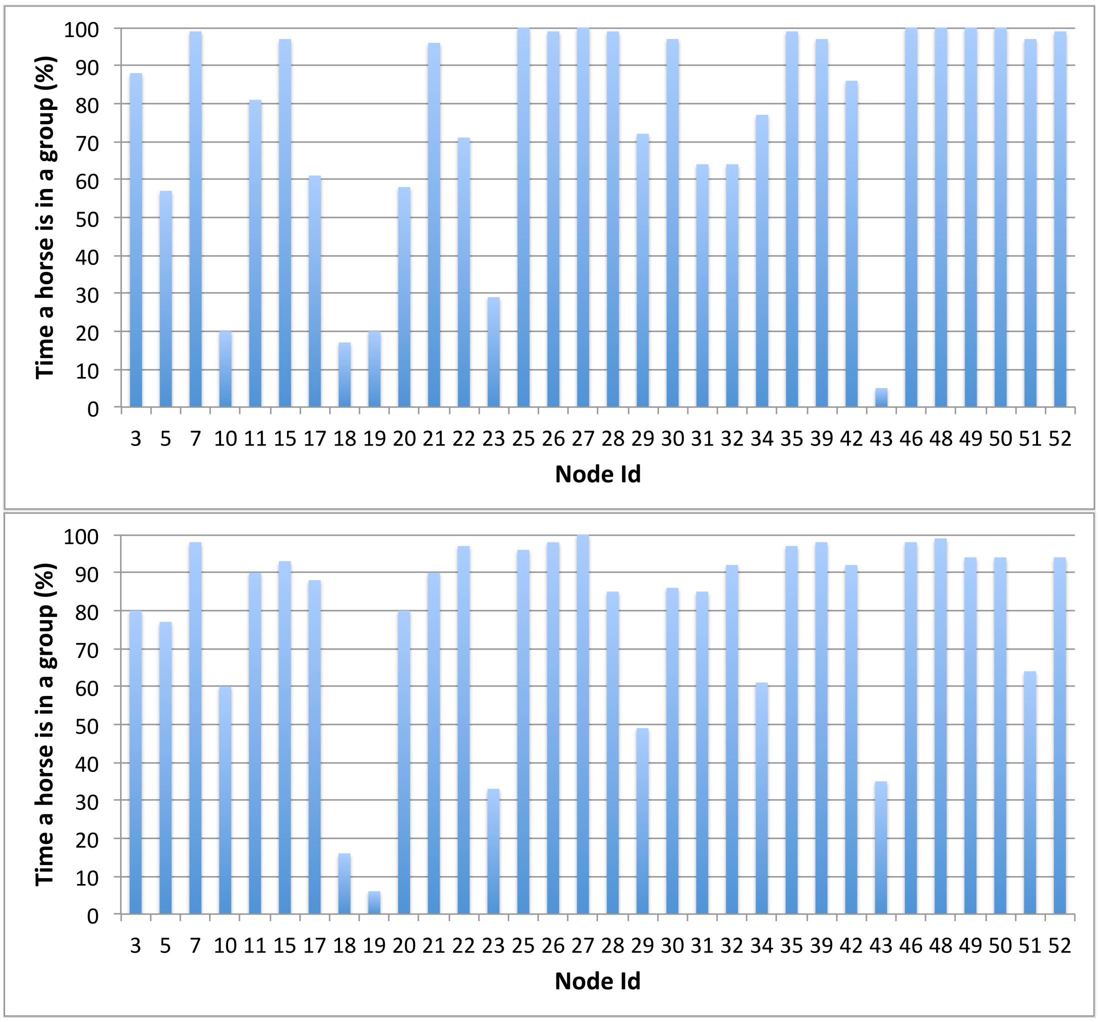


Figure 4.13: The percentage of time during the two-month (top) and nine-month (bottom) periods when a horse is part of a group.

cations on a map at the beginning of the deployment. Group 1 was the most stable grouping over the duration of the first two months, and is composed of one male (H7) and several females. We will follow Group 1 in more detail in Figures 4.16 and 4.17 to study the dynamic behaviour of this particular group. Figure 4.16 illustrates the size of Group 1 at hourly intervals and its variation over time, with eight horses being present in this group for a majority of the time. Figure 4.17 follows all the horses which had contact with Group 1 over the initial two-month period and shows some interesting dynamics. A core of seven horses - H7, H21, H39, H30, H35, H26 and H46 - remained together for the longest duration, with others dipping in and out. H48, for instance, started being part of this group, but detached after three weeks without re-joining for more than a few hours at a time (see Figure 4.18). The gaps in the graphs

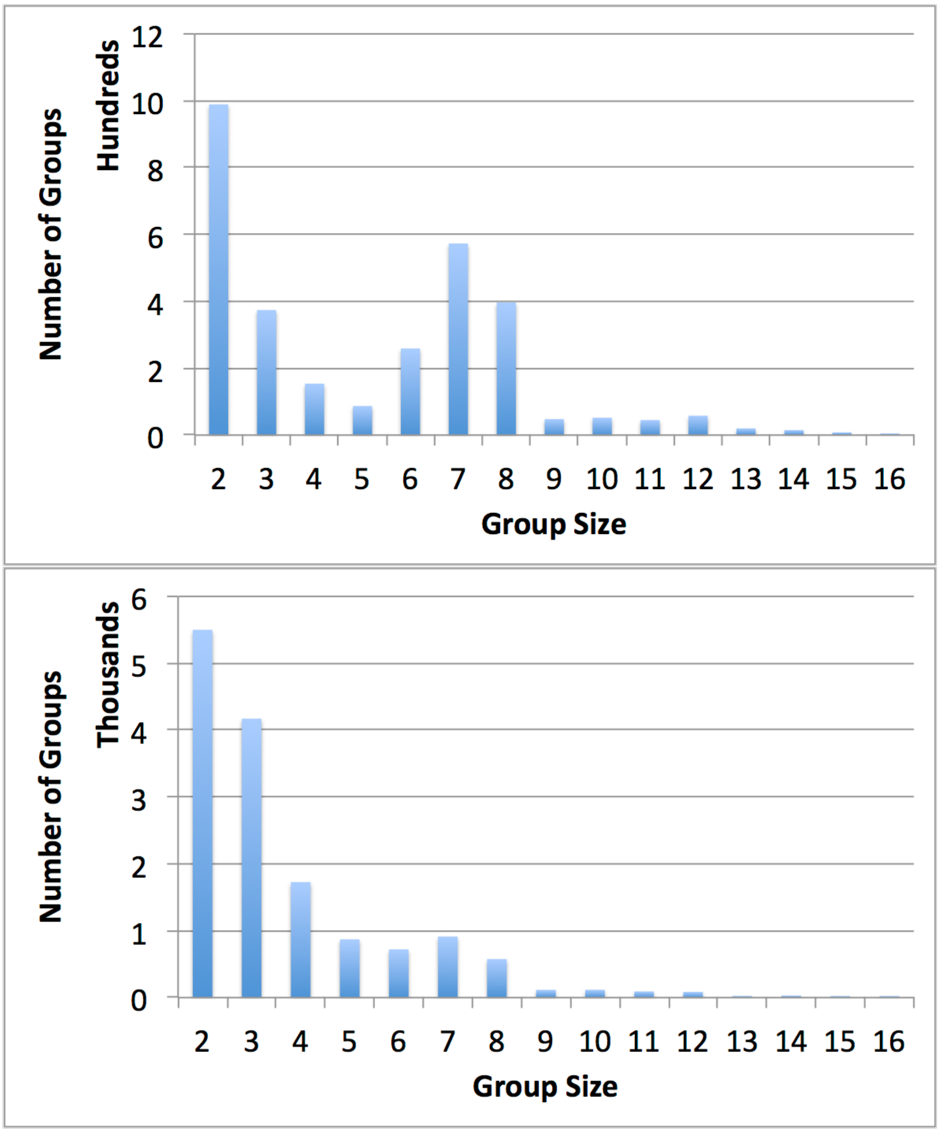


Figure 4.14: Instances of group sizes over the two-month period (top) and over the nine-month period (bottom).

for the core horses (Figures 4.16 and 4.17) indicate the period when one or more of the core horses left the group, even though other horses might have joined the group to increase its numbers. The graph also reveals that horse H7 was the dominant male when other male horses tried to join this group (H5, H10, H18, H19, H22, H29), as they only joined for very short periods of time before probably being chased away.

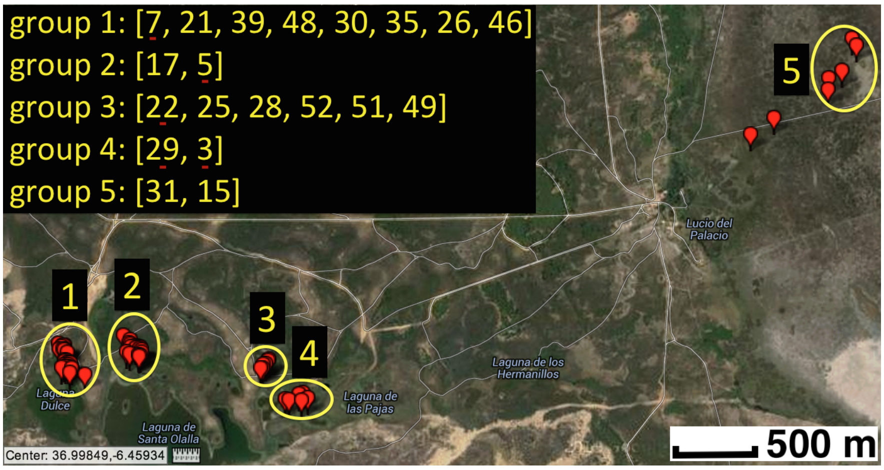


Figure 4.15: A snapshot of the groups with their locations in the first hour on 19 September 2013 (Note: each horse can have up to three locations in each hour).

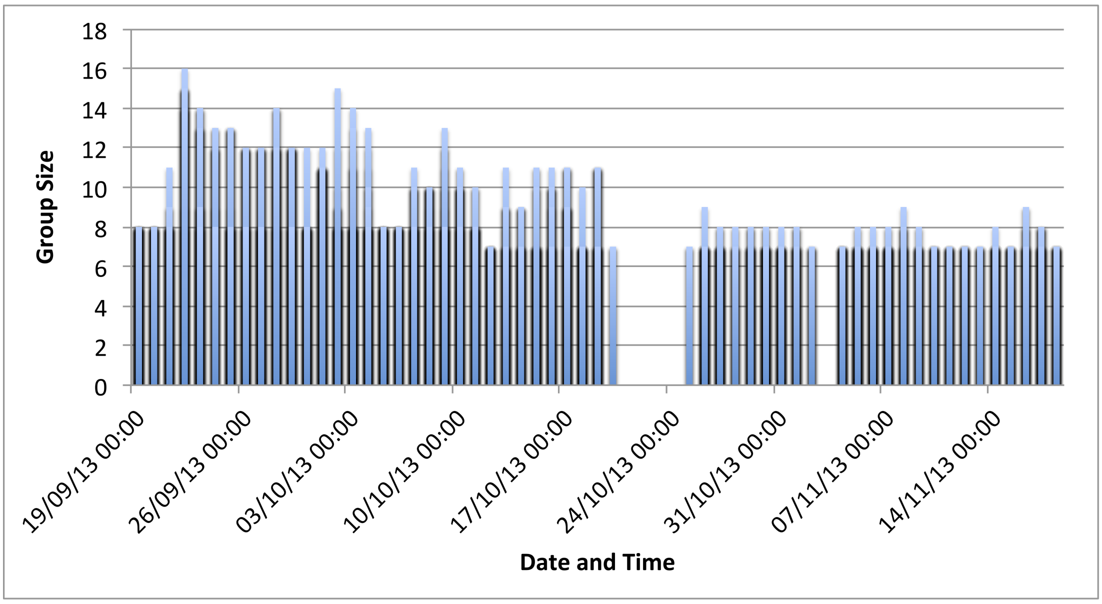


Figure 4.16: Group 1 size changes over time.

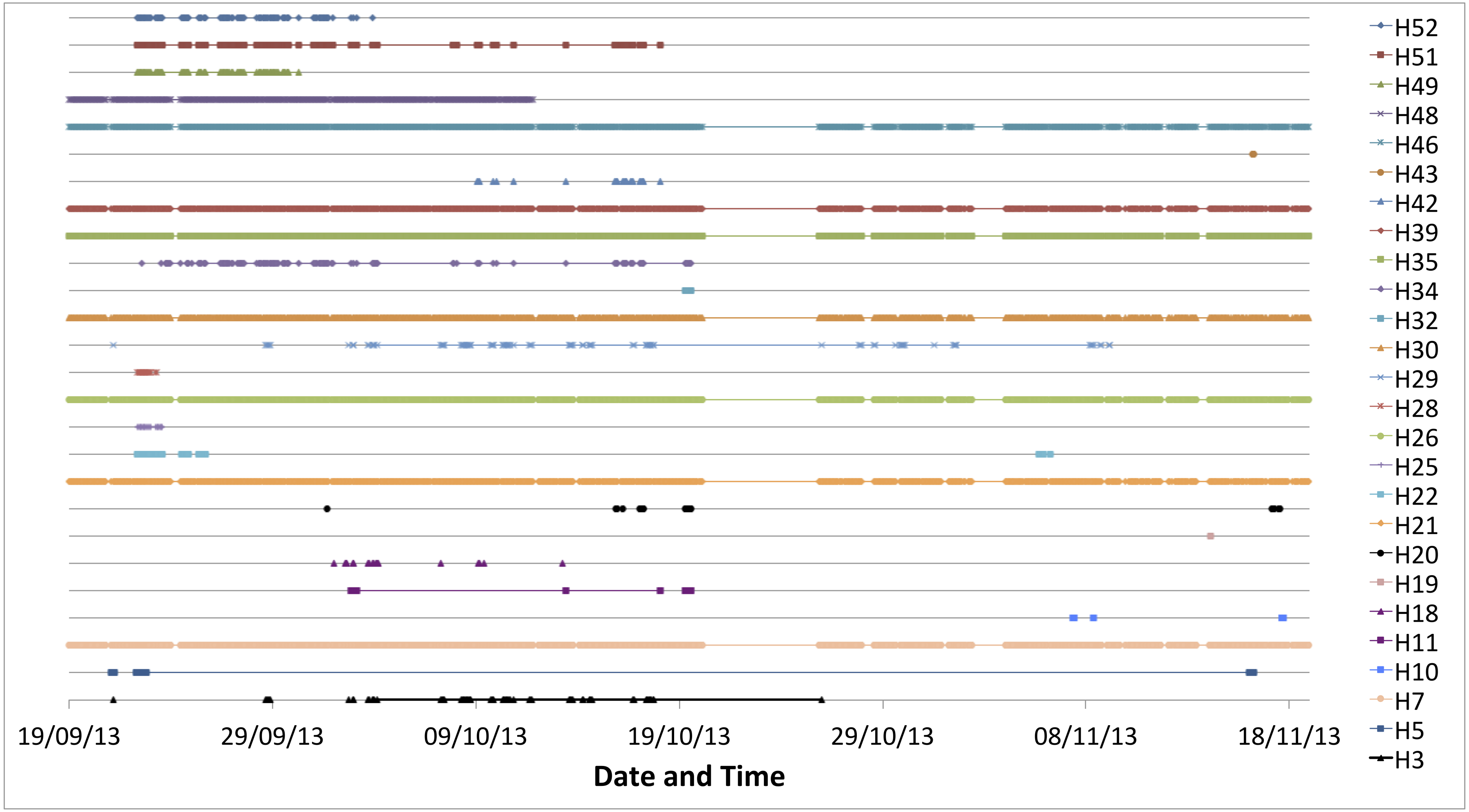


Figure 4.17: Dynamics of Group 1 over two months around the core horses - H7, H21, H39, H30, H35, H26 and H46.

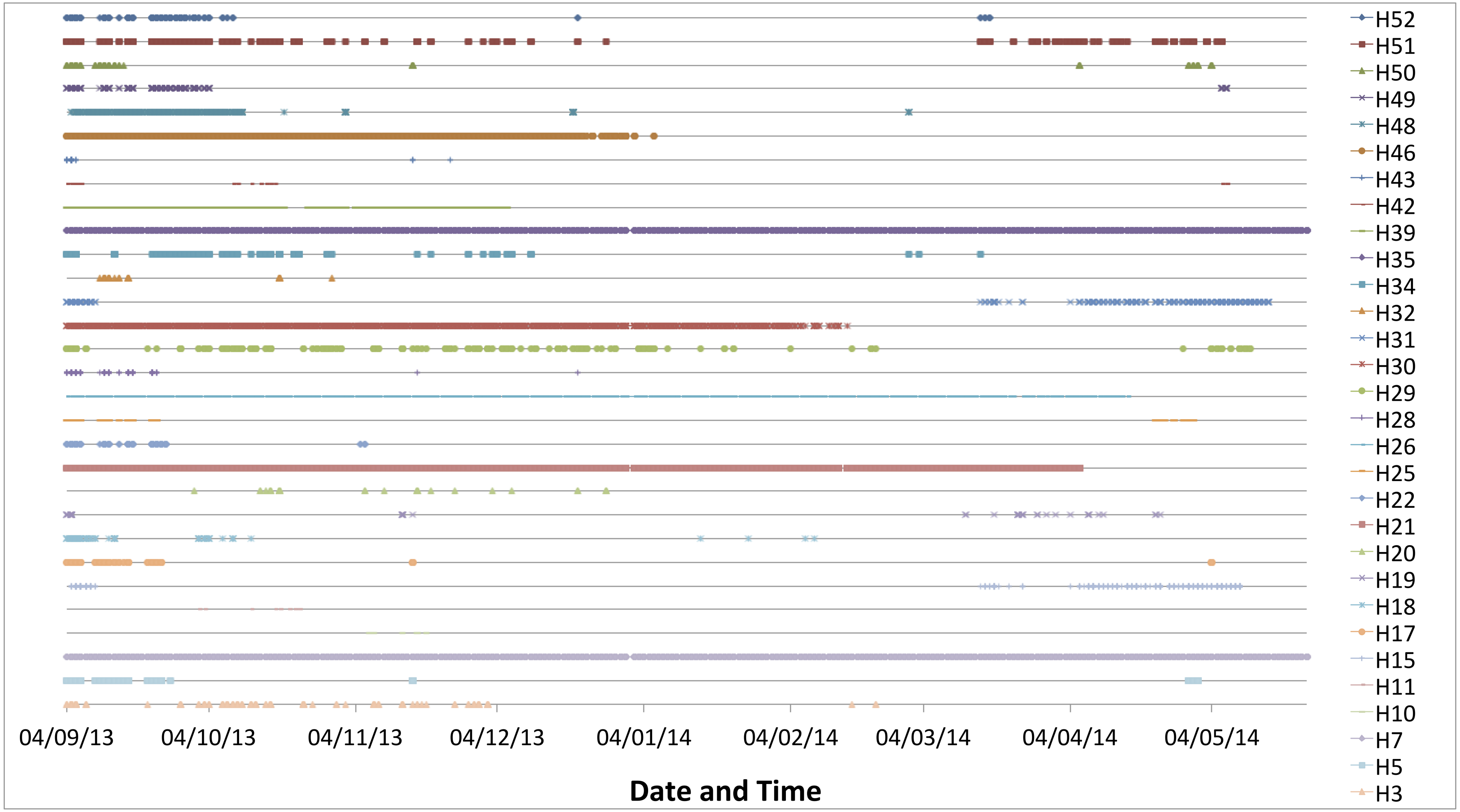


Figure 4.18: Group dynamics over nine months around the core horses - H7 and H35.

The Doñana National Park has a Mediterranean climate, with hot and dry summers, and cool and wet winters. Individual horses' movements over the nine months, from September to May, were considered. The movement patterns of horses belonging to different groups, as well as of the ones belonging to the same group, were observed, ensuring that the selection of results presented are representative. Figure 4.19 shows the monthly locations of horse H35 over nine months (September to May) and the locations in the first few days of June. It also shows the average temperature, the number of days with precipitation and the distance travelled by the horse in each of these months. There are significant correlations ranging from 0.54 to 0.76 between the distance travelled, the size of the area covered by the horses movements, and the average temperature and the number of days with precipitation.

Based on Figure 4.19, November seems to be the month when the transition from the dry season to the wet one is taking place in Doñana, assuming that floodings are the reason behind the observed limitations in the horse's movement. Some parts of the Doñana Park get completely covered by water, but even in the areas that do not, the movement of the animals may still be restricted. Figure 4.20 highlights the differences between the wet and dry seasons using both pictures and landsat images, along with adding the locations of horse H35 on the left-hand side. It can be observed that the horse is not situated in the area that gets covered by water during the wet season, and we are left to assume that its movements are constrained by less major floodings than the ones visible in the landsat images.

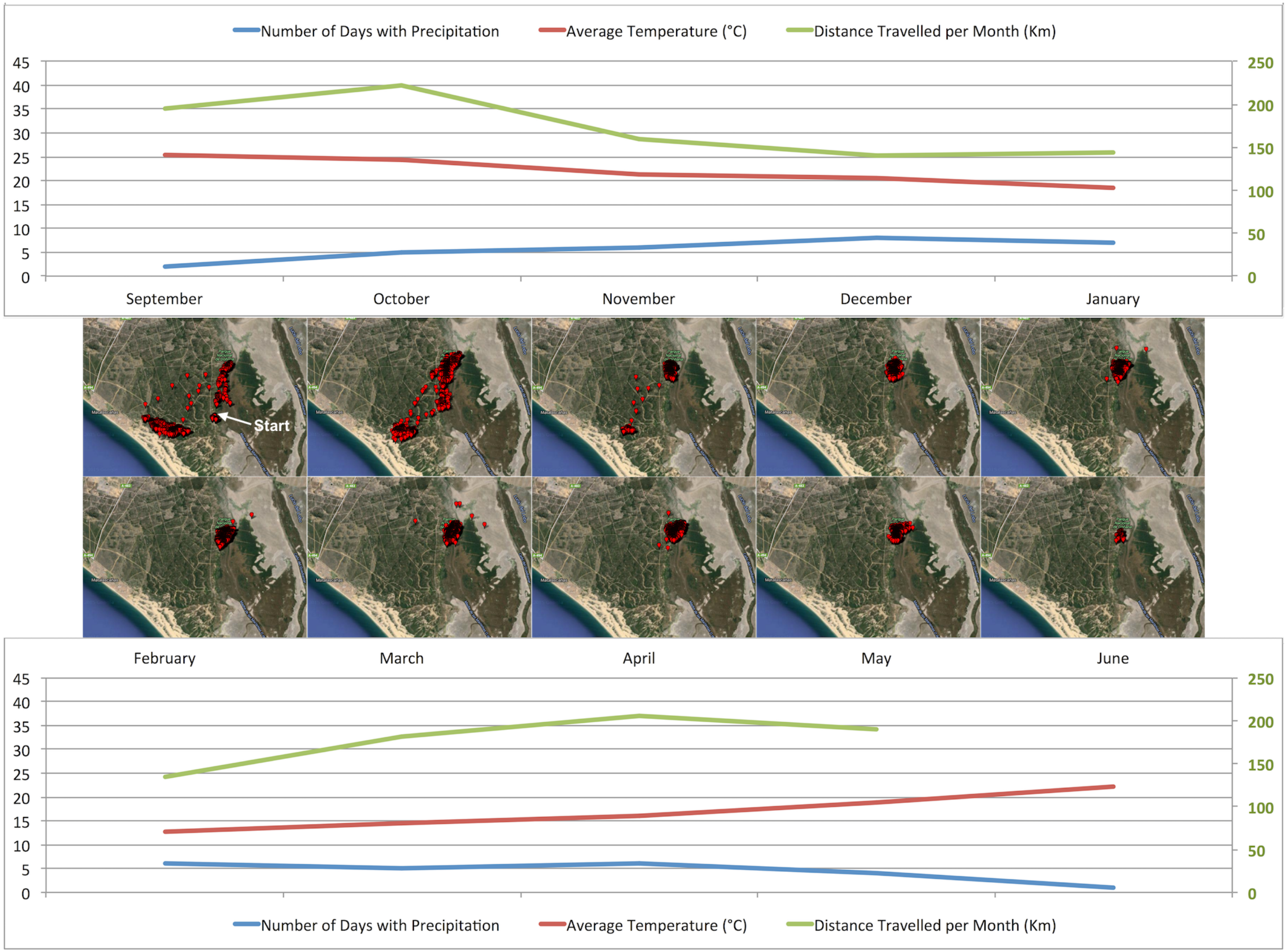


Figure 4.19: Seasonal variation: The locations of horse H35 over nine months from September to May and the first few days of month June (middle). The average temperature, the number of days with precipitation and the distance travelled by horse H35 (top and bottom).

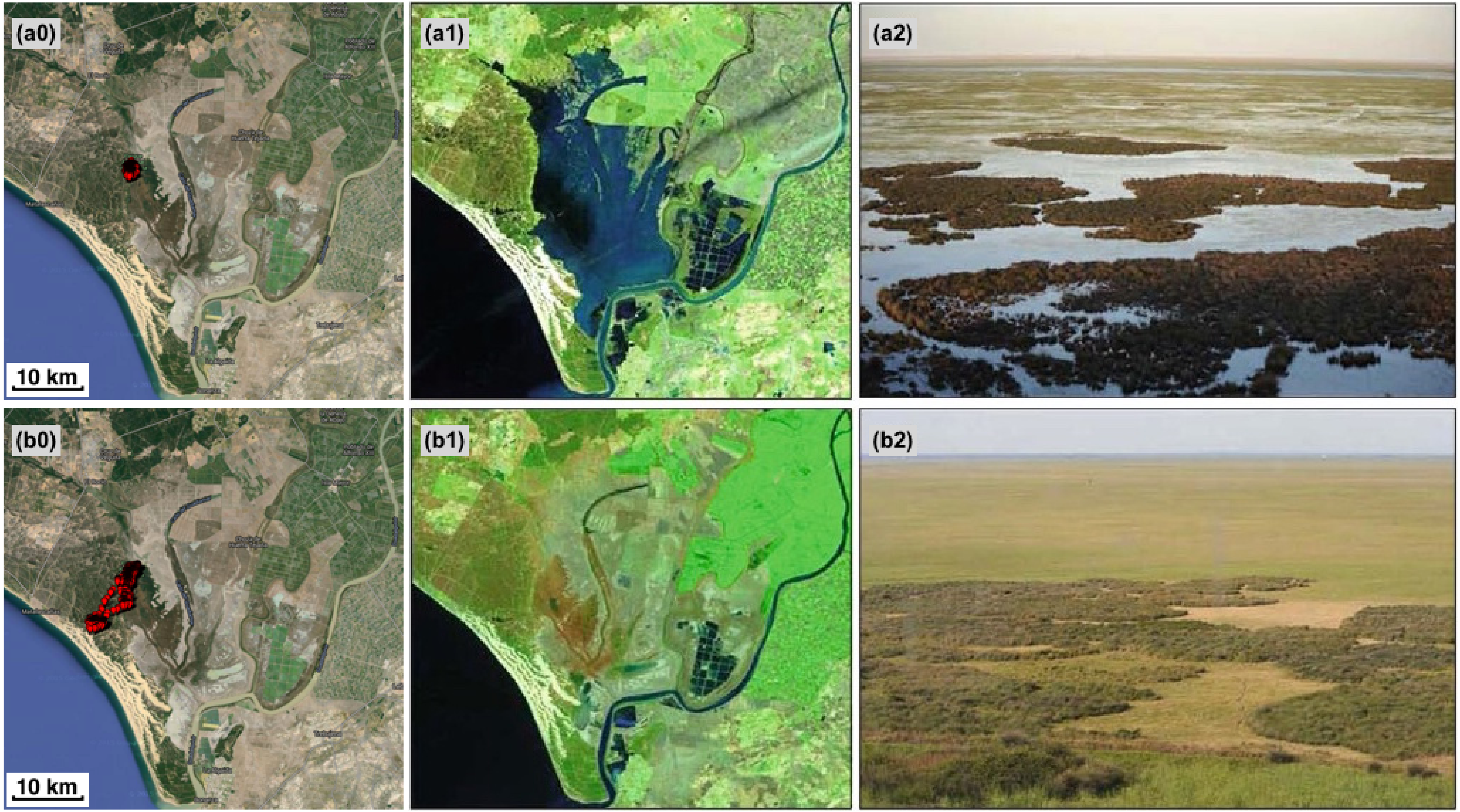


Figure 4.20: The locations of horse H35 in the months of October (b0) and December (a0). General (a1, b1) and local (a2, b2) views of Doñana marshland during wet and dry seasons. Landsat images elaborated by LAST-EBD (CSIC) (photos by C. Finlayson).

This project has been an exemplar for data-driven approaches to conservation by understanding animal behaviour with a level of spatial and temporal resolution than hitherto possible for a deployment over nine months. The insights obtained on the horses' behaviour could only have been possible thanks to a wireless sensor networks-based solution. Other current methods using satellite-based remote sensing and visual observations do not offer the same level of discrimination and efficiency. Our approach can have a considerable impact in animal behavioural sciences by extending to other species and environments where visual observation methods are currently used.

4.3 Short-term deployment with power-constrained base-stations

The VB-TDMA protocol was developed to extend the lifetime of the battery-powered base-stations in the absence of mains power supply in the deployment area. The testing of the protocol was conducted over a seventeen-day period with one base-station and eight mobile nodes attached to domesticated horses, part of a teaching herd at the School of Veterinary Studies, University of Edinburgh. This deployment had the role of testing the VB-TDMA algorithm under realistic conditions.

Our stated goal for this scenario was to obtain GPS data once every approximately 20-22 minutes in the wild, which translates to 23,890 samples over a twelve-month period for each mobile node. As the testing was intended primarily to exercise the VB-TDMA protocol, the data gathering was accelerated to once every minute over a seventeen-day period. The sensor node was able to record around 24,000 sets of sensor readings on a mobile node during the battery lifetime, which is more than the stated goal over the proposed twelve-month duration of deployment. The deployed VB-TDMA protocol implementation had mobile nodes attempting one discovery per sampling time slot, followed by possible upload slots, and data was uploaded on only one channel (one base-station was deployed).

There is an important caveat when projecting the performance over twelve months, based on the limited study. For a twenty-minute interval the ephemeris data (used to calculate the position of each satellite in orbit) and the almanac (information about the time and status of the entire satellite constellation) of the GPS modules will not be as fresh as in the case of one-minute intervals. Thus, the time taken to acquire a fix and to update the satellite information will be longer.

For a one-minute slot at an accuracy of less than two metres, the UC430 GPS module took about fifteen seconds on average to acquire and improve a position fix, at a cost of approximately twice the power consumption of the FGPMMOPA6H GPS module. The latter was active, on average, for 25.5 seconds every twenty minutes, having a lower, albeit acceptable accuracy of less than ten metres. Therefore, the overall power consumption of the UC430 aiming for a high accuracy fix every minute, is higher than the power consumption of the FGPMMOPA6H at a lower accuracy with a position fix every twenty minutes. We can therefore project that the lifetime of the deployment of nodes equipped with the FGPMMOPA6H GPS module, sampling roughly every twenty minutes, and aiming for an accuracy of less than ten metres, should be greater than one year.

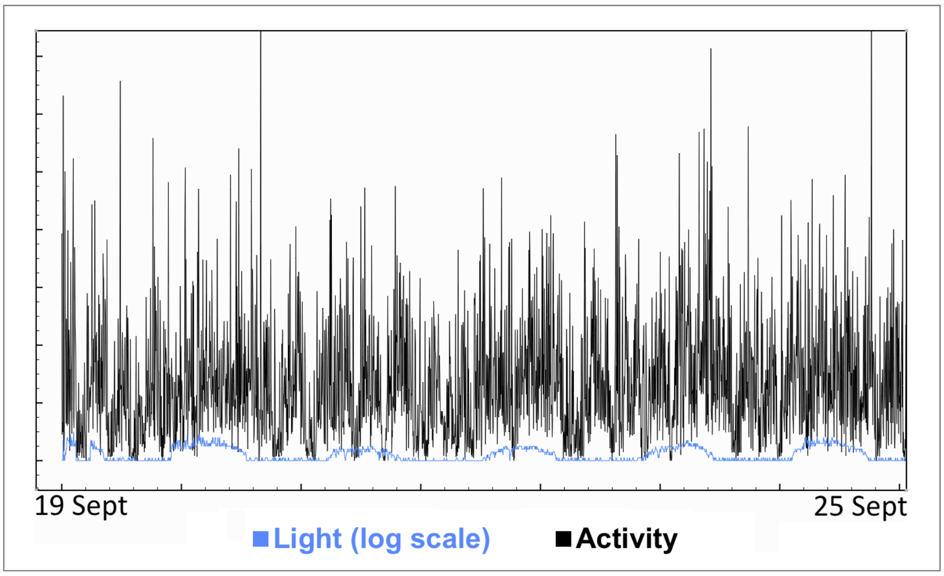


Figure 4.21: Activity and light over time.

Figures 4.21 and 4.22 display the light and activity measurements collected by the mobile node. Figure 4.21 considers a six-day slice of the deployment for better data visualisation. The light data is presented in blue in a logarithmic scale, whereas the activity levels of the horse are presented in black in a linear scale. The diurnal cycles can be easily identified in the graph. In both figures it can be observed that there are increased levels of activity during periods of light, which has been confirmed in the Doñana wildlife deployment in Figure 4.10. Also, there is a positive correlation of 0.63 between the hourly average activity levels and the light intensity over the entire duration of the deployment, which can be visualised in Figure 4.22. One can exploit the periods of inactivity to lower the frequency of certain sensor samplings in the effort

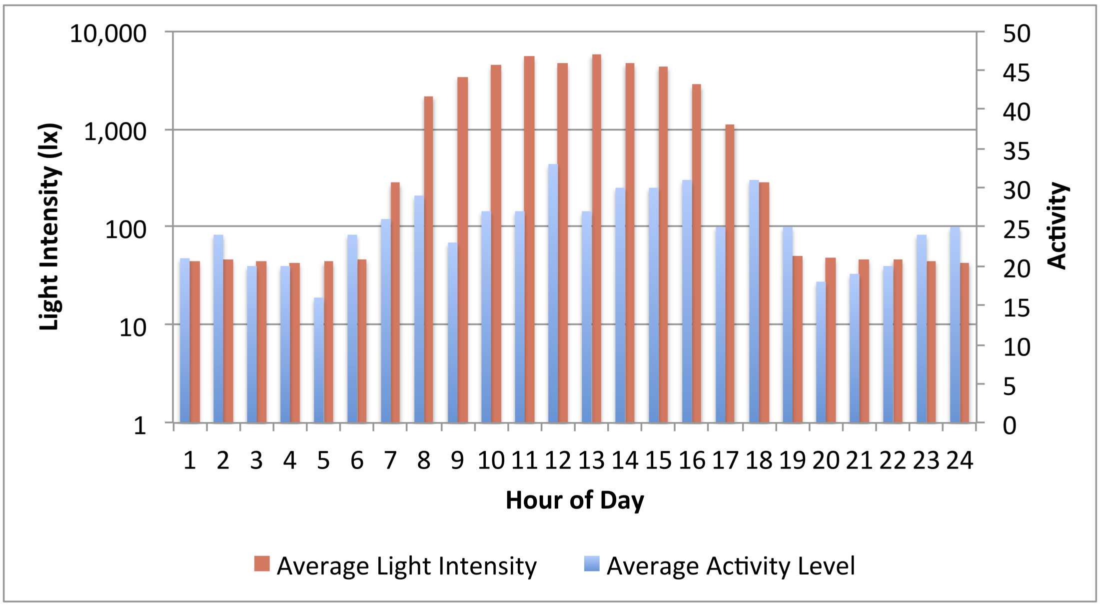


Figure 4.22: Hourly average of activity level (linear scale on the right-hand side) and light intensity (logarithmic scale on the left-hand side).

to extend the battery lifetime.

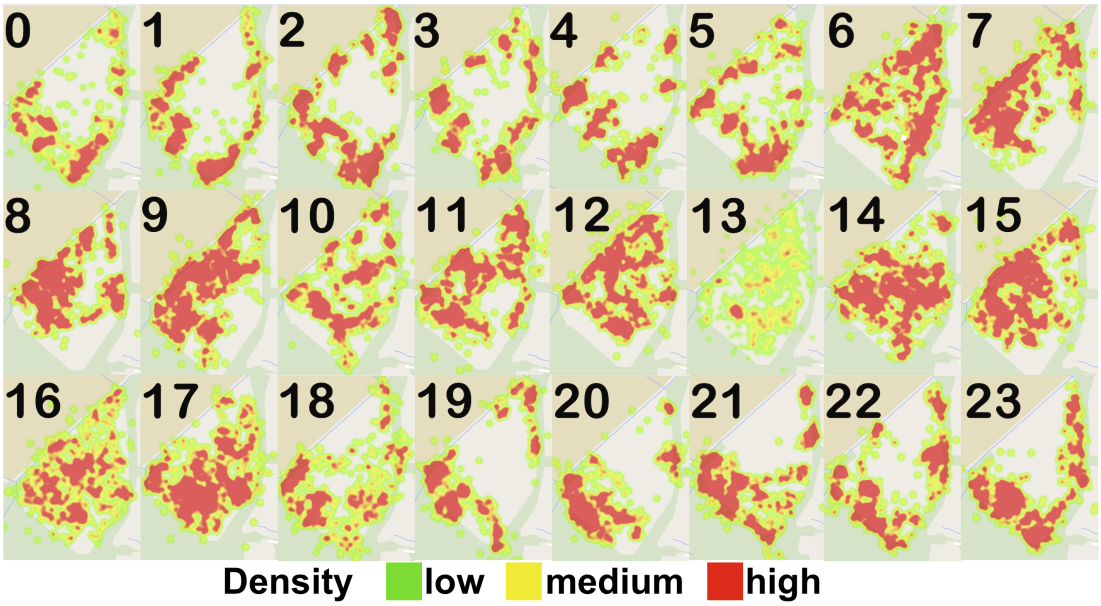


Figure 4.23: Heatmaps of the hourly snapshot of a horse's location aggregated over 17 days.

Figures 4.23 and 4.24 give an interpretation of the horses' activity and behaviour during the diurnal cycle based on the GPS location data. It is interesting to observe that during the night times the horses avoid the open field and prefer locations that

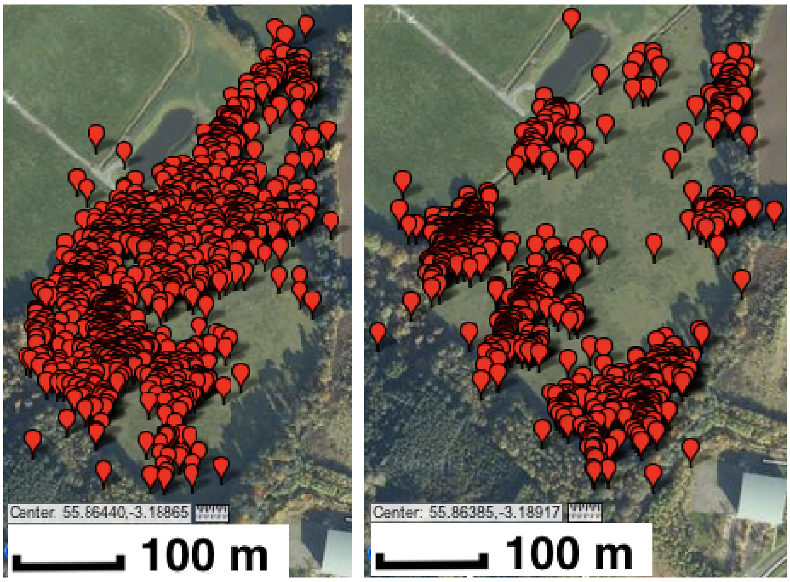


Figure 4.24: Typical day hour: 9am (left), and night hour: 4am (right).

are sheltered by trees along the periphery of their confinement area. This behaviour is noticeable in Figure 4.23 in the hourly snapshots of a horse's locations aggregated over the seventeen-day deployment period, as well as in Figure 4.24 which shows on the left-hand side the positions of one horse gathered over seventeen days between 09:00 to 09:59, and on the right-hand side between 04:00 to 04:59. It was also observed that the horses move less during night times. During a typical night hour, a horse was located inside tighter circles with an average radius of 23m, whereas during the day-time the average radius was 70m.

4.4 Summary

The requirements of the scientific study of animal behaviour, the welfare of the rare breed of horses, and the harsh environmental conditions, make the scenario of long-term tracking and monitoring of wild horses highly challenging. This chapter demonstrated the complete solution on this representative scenario. The network behaviour data collected from the deployments plays an important role in relation to the simulation models, as it provides the basis for their validation and it is used to enhance their accuracy in reflecting the reality. The sensor data analysis offers insights into the horses' individual and group behaviour, thus demonstrating the behaviour monitoring capabilities of the proposed solution. The discussion of the trade-offs and the final design choices, justified based on evidence from the experimental studies, may prove valuable to someone designing a similar solution.

The next chapter makes use of the data collected from these deployments to validate simulation models corresponding to the hardware platform and communication algorithms. Furthermore, real data is injected into simulations to offer more realistic predictions of the network behaviour using the VB-TDMA protocol for long-term deployments, and comparing it to other low-power MACs.

Chapter 5

Model Validation and Performance Analysis in Simulations

The use of computer simulations is an established method for estimating the performance of networked systems. The system can be modelled at different levels of abstraction ranging from behavioural-level architectural models through to cycle-accurate models of hardware components. The development and evaluation of protocols and algorithms for networked systems can benefit significantly from a simulation environment for experimentation. Once the behaviour of the new protocols is validated, simulations offer the opportunity to evaluate and compare them with other solutions in a controlled environment. The insights, results and data from the deployments presented in the previous chapter are used to augment simulation models aimed at analysing the performance of our proposed solution. The simulation results presented in this chapter can be divided into three parts. The first part describes the validation of the hardware simulation models against results from the deployment in Doñana. The second part looks at the validation of the VB-TDMA protocol's implementation in simulation. The third part provides a performance evaluation of the VB-TDMA protocol, including comparisons to a selection of existing low-power MACs. The access to real data from the deployment, on the movement of the horses and the time taken by the GPS to acquire position fixes, has enhanced the models for simulating network behaviour more realistically and estimating the power consumption of the nodes. Complete sets of data from ten horses over a period of forty-four days were selected to feed into the simulation models (mobility and GPS). The representative movement behaviour over the forty-four days was used to make predictions over an extended period of twelve months using the data-mirror mobility model introduced in this chapter. The network behaviour provided by the data-mirror mobility model was validated against real movement data over a six-month period, to account for seasonal variations.

5.1 Hardware Models Validation

5.1.1 Hardware Models Validation Scenario

The validation of the simulation models is essential to give confidence in their ability to reflect the real world. The period was chosen to be feasible as a test pre-deployment. The hardware models (introduced in Section 3.2.1) are validated for a representative scenario for the targeted class of applications by comparing the simulated network behaviour to the one obtained during deployment. Any discrepancies in the network behaviour results in the validation process need to be justified by determining the reasons for the differences. For example, the main cause of discrepancies in network behaviour for the first forty-four days of the Doñana deployment can be explained by the unreliability of the power supplies for the base-stations, which caused down-time periods that could not be simulated as these episodes were not logged.

The hardware simulation models were validated using the data collected over the initial forty-four days of the Doñana deployment. The reasons for choosing this period were both practical and technical. The period is short enough to be feasible as a test pre-deployment for collecting movement information, and large enough to capture representative movement behaviour over a period of twelve months and to facilitate the experiments presented in Section 5.1.2. Also, the majority of the collars were uploading data during this period (see Figure 4.9), and the effects of base-station down-times were minimal at the beginning of the deployment.

The simulations were supplemented with inputs from the real-world deployment: the GPS coordinates and the speeds of travel between consecutive locations were imported into the mobility model, and the times to acquire position fixes were used by the GPS model. The mobility data offers accurate network behaviour in terms of the base-stations' data collection patterns and data upload latencies, while the time when the GPS was turned on represents the main source of energy consumption. Furthermore, the delays due to the staggering of the commissioning of the base-stations over a period of few weeks following marking the horses with mobile nodes, were accounted for in the simulations.

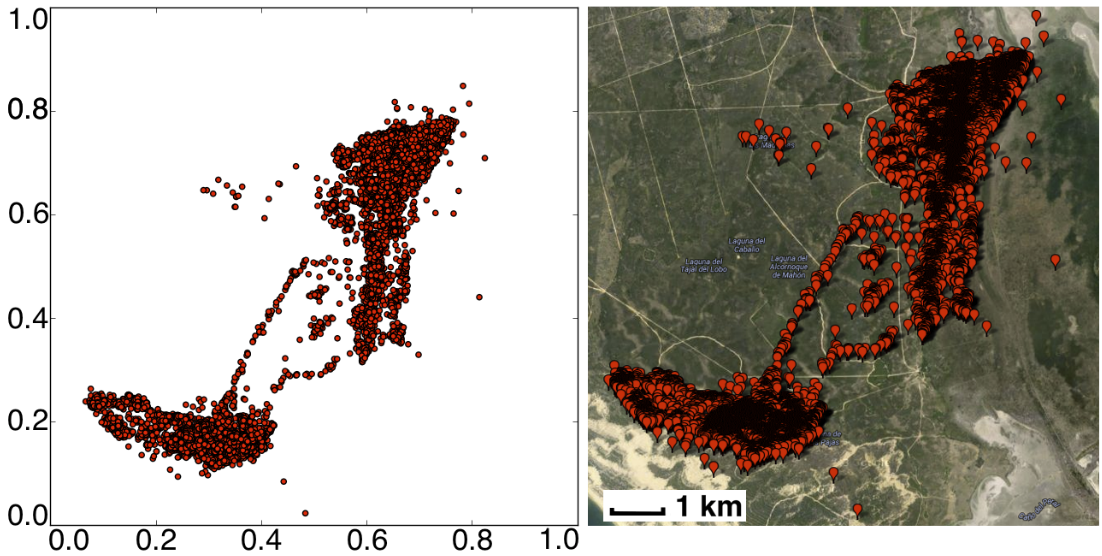


Figure 5.1: Converted locations in a plane - matplotlib (left); GPS locations on map - Google Maps (right).

5.1.1.1 Conversion of GPS Locations to 2D Coordinates

The locations gathered in the deployments needed to be converted to corresponding positions within the simulation area in order to map the movement of the mobile nodes. Replicating the deployments in as much detail as possible is important for the validation process. As described in Section 3.2.3.2, due to the ellipsoid shape and the curvature of the Earth, the haversine formula [107] was used to convert the GPS coordinates into coordinates within the SpeckSim simulator. Figure 5.1 illustrates the fidelity of the conversion method that uses this formula for transforming the GPS coordinates from the deployment to the planar surface used in the simulations. The visual validation provided by this figure offers a satisfactory accuracy for the simulated scenario. It should be noted that the data presented in this figure belongs to a larger dataset than that of the ten horses over forty-four days used in the simulations.

5.1.1.2 Amount of Data Uploaded

The amount of data collected by the network consists of the number of unique packets uploaded and the redundant ones. The emphasis is on the unique data, as the redundancy only serves as a backup. Although the patterns of the spread of the uploaded data over the base-stations look similar in Figure 5.2 (the same base-stations collected packets in reality and simulation, with B2 collecting the most, followed by B1 and B5), the simulations exhibit higher levels of redundancy than evidenced in the real

Table 5.1: 44-Day Validation Scenario - Received Packets

Metric	Reality	Simulation
Total number of received packets (including duplicates)	31757	52584
Total number of unique received packets	30002	31072
Redundancy percentage	5.85%	69.23%

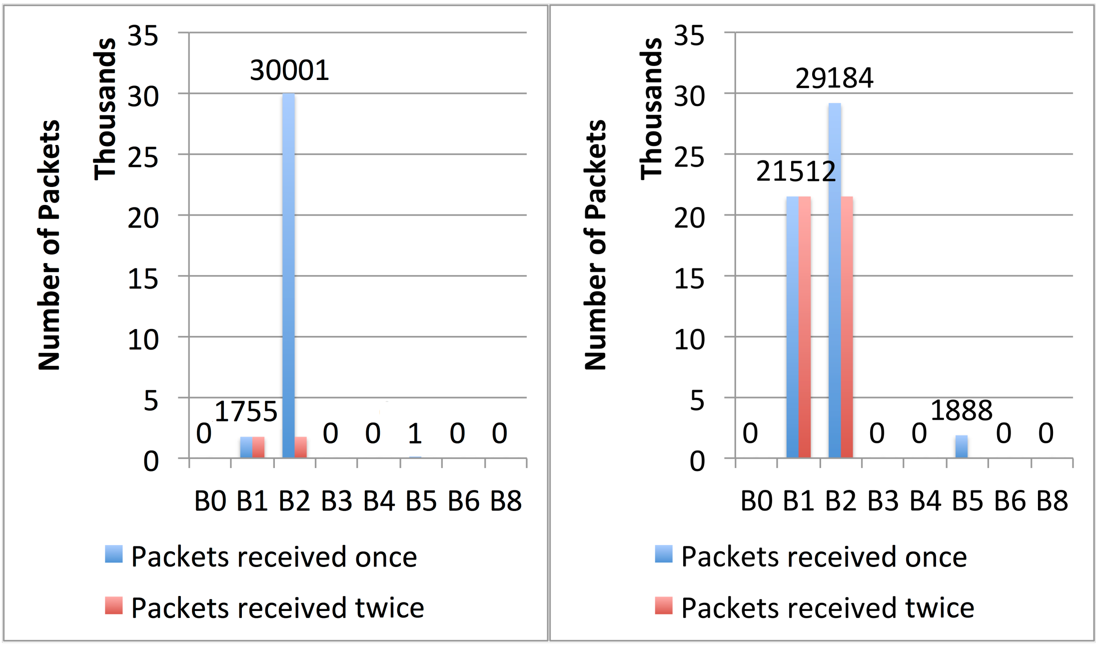


Figure 5.2: Number of uploaded packets and redundancy: real deployment (left), simulation (right).

data. Approximately 65% more packets were collected in the simulation out of which 94.8% were duplicates. This is due to the downtimes of the base-stations in the real deployment. It also explains why B2 received more packets in reality than in simulations (see Figure 5.2). This is further explained and validated in Section 5.1.1.3. The more meaningful result is that the difference between the number of unique packets received in simulations and in reality is only 3.5%, as a result of the better operation and reliability of B2 (due to having a more reliable power supply).

The skewing of the upload pattern towards base-station B2 reflects the reality of the horses being in the vicinity of this base-station for a majority of the time.

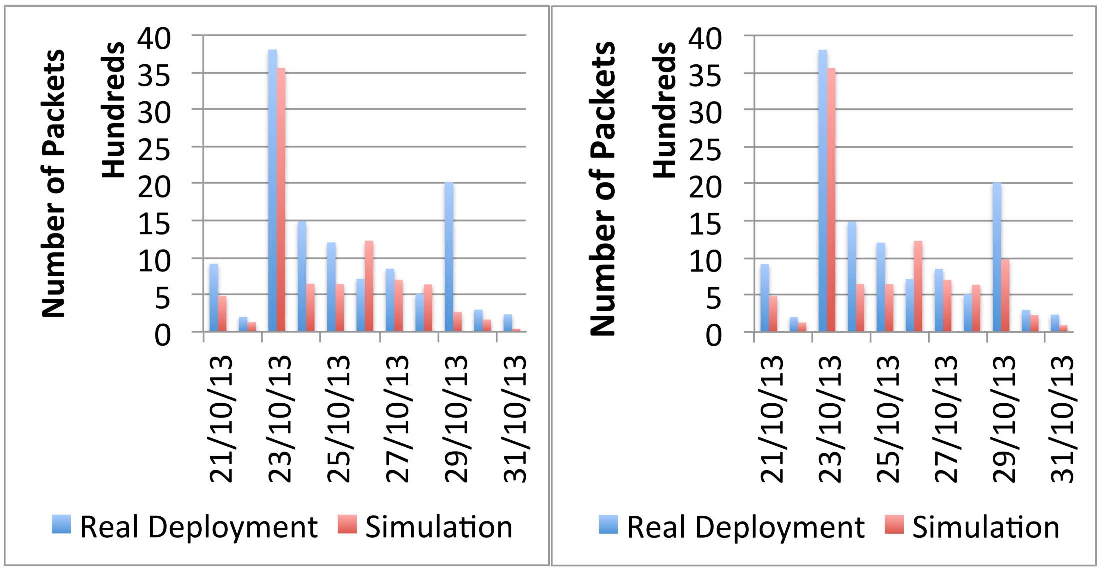


Figure 5.3: B2 upload pattern for an eleven-day segment: B5-on (left) and B5-down (right).

5.1.1.3 Base-station Upload Pattern

We considered the impact of the presence and the absence of base-station B5 on the hourly and daily packet upload patterns for B2. Figure 5.3 presents on the left-hand side the daily pattern for a period of eleven days for both the real deployment and the simulation. The correlation calculated for the hourly upload pattern is 0.57, and for the daily upload pattern is 0.84. These correlation figures give confidence in the ability of the platform simulation models to replicate the trends in the network behaviour based on real mobility data.

An explanation for B2's lower number of packets collected in simulations is that in reality B5 probably had downtimes when the horses were within its range, accounting for B5's lower number of collected packets compared to the simulations. B5 was first deployed during the period presented in Figure 5.3, which was also reflected in the simulation. Since B5 and B2 are on the same channel, the packets collected by one will not be collected by the other.

In order to test the impact of B5's downtimes during this period, another simulation was run without activating B5. The effect of this on B2's upload pattern is presented on the right-hand side in Figure 5.3. B2 received more packets in the last three days presented in the graph and the correlation figure for the daily upload pattern improved to 0.91. The remaining discrepancies in the data upload pattern may be explained by B2's downtimes. If a base-station does not acknowledge the packets sent by the mobile nodes, the queue for that channel will grow and when the base-station is next active it will receive more packets, including the ones gathered during its downtime period.

5.1.2 The Data-Mirror Mobility Model

Mobility models characterise the movement of mobile entities, and how their locations and speeds change over time. They are used to simulate and evaluate the performance of mobile wireless systems, and the algorithms and protocols at their basis.

We propose the data-mirror model, which is an instance of a data-driven model. The virtue of this data-driven model is that it is effective yet simple for the case in point: a sliver of real movement data (containing the locations that a node has to pass through and the speeds at which it moves between any two locations) is concatenated to its mirror image (representing the inverted path); this process is repeated recursively for the length of deployment. Thus, for a twelve-month simulation, the mobile nodes are going back and forth on the routes determined by a slice of forty-four days of real movement data. Whether this will generalize to other scenarios is unproven but does not detract from the results presented. The intuition underlying this model is that having a structure reflecting the reality of horses' movements is better than relying on heuristics used in trace-driven models.

We also developed a trace-based mobility model (refer to Section 3.2.2 for details) for comparison. Trace-based mobility models exploit measurements to generate synthetic traces based on probabilities derived from real data. In this case, the distances and speeds of the nodes are generated so that their average matches the average distance travelled by a horse and the average speed at which the horse travels during twenty-minute intervals calculated from the real data. The selected standard deviations match the spread of the speeds and of the distances from the real data. We demonstrate that compared to this trace-based mobility model, the simulated network behaviour using the data-mirror mobility model better emulates the simulation results in the context of network behaviour. This is done by comparing the results from these two movement models to a reference case. Furthermore, the data-mirror mobility model's ability to predict network behaviour in the face of seasonal variations was validated over six months.

The reference case is similar to the validation scenario from Section 5.1.1, but with the base-stations running from the beginning of the deployment and without any downtime periods. The movement of the mobile nodes in simulations replicates the exact movement of the horses in the nature reserve over a forty-four-day period (which is one eighth of a full one-year deployment period). The data-mirror model was tested for three different scenarios: one-half, one-quarter and one-eighth slices of the 44 days movement data. The simulations for these three scenarios were compared against the reference case using the real movement data over the forty-four days.

Another eight simulations were performed using the trace-based mobility model based on random direction. Each simulation uses a different seed for the random method. These simulations are also compared to the reference case with the results presented in Table 5.2.

Table 5.2: Mobility Models Comparison

<table>
<tr><th>No</th><th>Metric</th><th>Reference</th><th>Half:
22-day slice</th><th>Quarter:
11-day slice</th><th>Eighth:
5.5-day slice</th><th>Trace-based
mobility</th></tr>
<tr><td>1</td><td>Average correlation of the packet upload distribution over the base stations | Standard Deviation</td><td>-</td><td>0.95 | 0.02</td><td>0.72 | 0.33</td><td>0.62 | 0.41</td><td>0.32 | 0.16</td></tr>
<tr><td>2</td><td>Correlation of the average packet upload distribution over the base-stations</td><td>-</td><td>0.99</td><td>0.98</td><td>0.96</td><td>0.3</td></tr>
<tr><td>3</td><td>Average percentage of battery left | Standard Deviation</td><td>89.1</td><td>89.02 | 0.07</td><td>88.95 | 0.17</td><td>88.97 | 0.15</td><td>88.7 | 0.14</td></tr>
<tr><td>4</td><td>Average total number of uploaded packets | Standard Deviation</td><td>95009</td><td>93500 | 5496</td><td>85009 | 25680</td><td>81078 | 24319</td><td>19813 | 11100</td></tr>
<tr><td>5</td><td>Average total number of unique packets uploaded | Standard Deviation</td><td>31231</td><td>31674 | 6</td><td>31666 | 23</td><td>31560 | 131</td><td>11633 | 3989</td></tr>
<tr><td>6</td><td>Average network latency | Standard Deviation</td><td>01:10:03</td><td>01:47:25 |
00:36:00</td><td>02:32:09 |
2:18:51</td><td>02:34:02 |
02:17:53</td><td>7 days 12:17:17 |
2 days 15:32:39</td></tr>
<tr><td>7</td><td>Average redundancy percentage 2-level | Standard Deviation</td><td>1.83</td><td>0.43 | 0.3</td><td>1.89 | 1.2</td><td>3.48 | 4.7</td><td>19.38 | 12.6</td></tr>
<tr><td>8</td><td>Average redundancy percentage 3-level | Standard Deviation</td><td>72.89</td><td>65.52 | 21.5</td><td>60.83 | 34.2</td><td>62 | 38.2</td><td>17.71 | 13.8</td></tr>
<tr><td>9</td><td>Average redundancy percentage 4-level | Standard Deviation</td><td>18.86</td><td>21.23 | 20.2</td><td>14.94 | 18.9</td><td>9.8 | 16.1</td><td>0.96 | 2.2</td></tr>
</table>

In the case of all nine metrics for evaluating the network behaviour, the data-mirror mobility model better tracks the reference case compared to the trace-based model. The fidelity of the model drops as the time slices become smaller, as would be expected, but even with the smallest slice chosen (one-eighth), the performance is better than the trace-based model. However, as the time-slices get narrower they are less representative of the movement patterns, and will also fail to capture seasonal variations over a year.

For testing the impact of the data-mirror mobility model on the network behaviour while accounting for seasonal variations over six months, the following validation scenario is considered. Two six-month deployments of six mobile nodes are simulated, one using actual movement data captured from the real deployment, and another mirroring the initial 44-day slice of this data for obtaining a six-month movement model.

The results are presented in Table 5.3 and Figure 5.4. The patterns for the distribution of the packet collection over the base-stations are similar. This is confirmed by the high correlation of 0.88. The differences for the average battery levels at the end of the deployment, the total number of packets and number of unique packets collected are very small, as can be seen in Table 5.3. The distribution of the redundancy levels also shows a strong correlation of 0.99, and the average network latency differs by only 6%.

These comparisons with the real movement data over the six-month period validate the data-mirror mobility model as a proxy for analysing the network performance of the mobile ad hoc network in simulations.

5.1.3 Twelve-month Prediction

Twelve months is usually the preferred length for wildlife tracking and monitoring deployments, as it is long enough to include all seasonal variations. Moreover, the duration also matches the permitted frequency for capturing the Retuerta horses in order to collect blood samples and give them their annual vaccines.

The results presented in this section are based on predictions over twelve months for three scenarios: an ideal scenario in which all the base-stations are activated from the start of the deployment without any downtimes; a best-case scenario where the nodes are always within range of a base-station; and a worst-case scenario where the nodes never get to empty any of their queues. All these simulations have the nodes running the asynchronous data upload protocol.

Table 5.3: Data-Mirror Mobility Model - Validation Over Six Months to Account for Some Seasonal Variation

No	**Metric**	**Movement based on real data**	**Movement based on mirrored data**	
1	Correlation of the packet upload distribution over the base stations	-	0.88	
2	Average percentage of battery left	53.67	53.28	Difference 0.7%
3	Total number of uploaded packets	233844	249891	Difference 6.8%
4	Total number of unique packets uploaded	78813	78840	Difference 0.03%
5	Average network latency	02:50:49	03:01:21	Difference 6.1%
6	Redundancy percentage 2-level	99.77	99.93	Correlation 0.99
7	Redundancy percentage 3-level	91.16	90.48	
8	Redundancy percentage 4-level	5.76	26.54	

5.1.3.1 Scenario with Ideal Conditions

By combining the hardware platform simulation models validated in Section 5.1.1 and the data-mirror mobility model presented in Section 5.1.2, this scenario aims to predict the network behaviour for a deployment length of twelve months under ideal conditions (having all base-stations running from the beginning of the deployment and without having any downtime periods). Figure 5.5 shows a high packet collection redundancy, with an average network latency of 2.5 hours. At the end of the deployment, all mobile nodes were still running and their average battery level was down to 8.24%, indicating that the nodes had battery to run for an extra 32 days and 18 hours.

The average lengths of the packet queues (each queue corresponding to one of the radio channels) of all the nodes during the deployment are: Q1:339, Q2:1121, Q3:18, Q4:9500. Figure 5.6 shows that Q4 is the largest for all the nodes and Q3 the smallest. This implies that out of all the chosen locations for base-stations, the locations for the

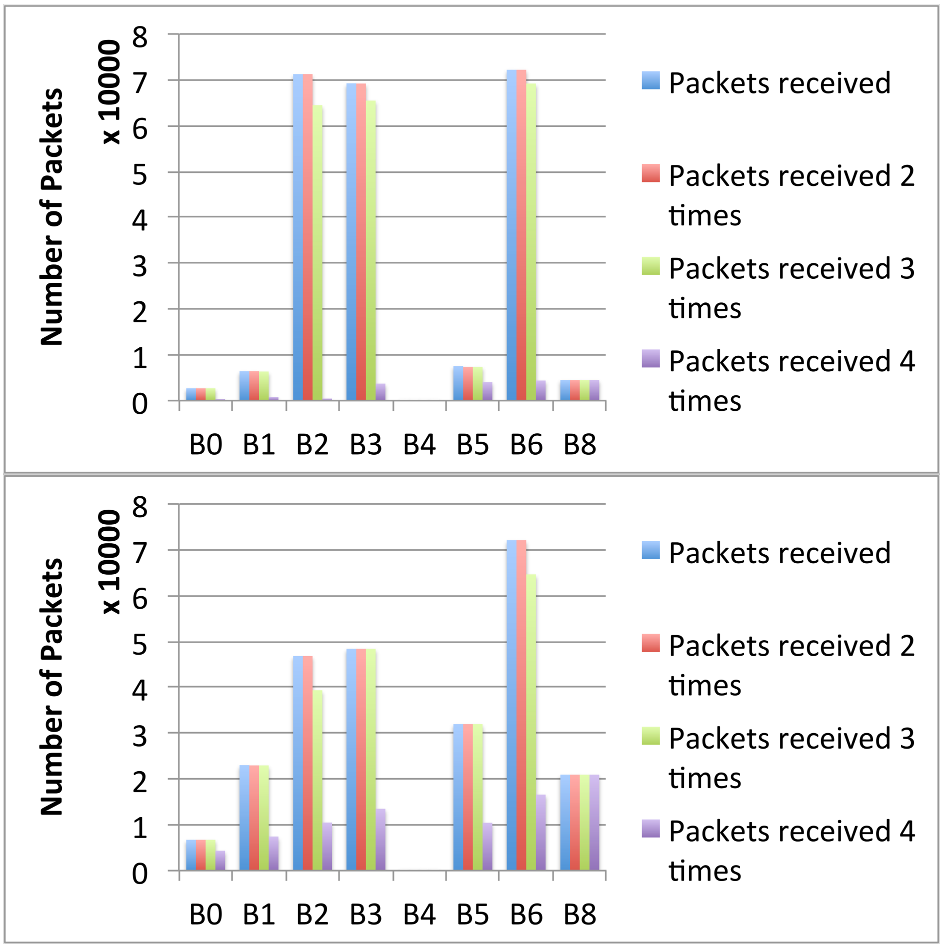


Figure 5.4: Six-month data collection spread over the base-stations: real mobility data (top) and mirrored mobility data (bottom).

base-stations on channel 3 are the best and the one for the base-station on channel 4 is not ideal for the movement pattern extracted from the 44-day sample.

For five out of ten mobile nodes, Q4 was never empty, meaning that the nodes were trying to upload a packet every fifteen seconds during the entire deployment, maximizing the power consumption of the radio for this channel. Q4 corresponds to the fast upload channel, which uses bursts of six packets. However, these bursts are dependent on acknowledgements, thus if a packet within a burst is not acknowledged, the burst will stop. Therefore, the power consumption on this channel, when mobile nodes are not in range of base-stations, is not greater than on the other channels due to these implementation differences.

Figure 5.7 presents the total number of transmitted packets for each node over the length of the deployment. Figures 5.8 and 5.9 show the queues lengths for node 1 (highest number of transmitted packets) and node 6 (lowest number of transmitted

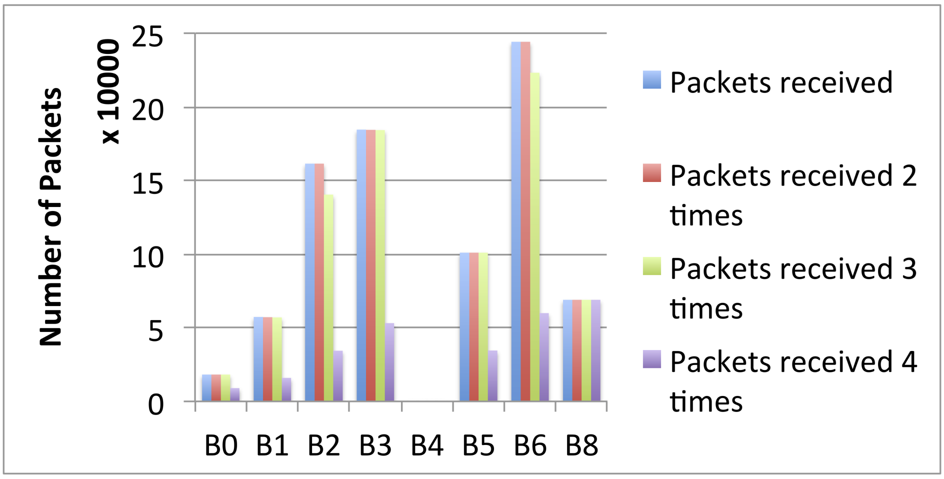


Figure 5.5: Network redundancy - twelve-month deployment.

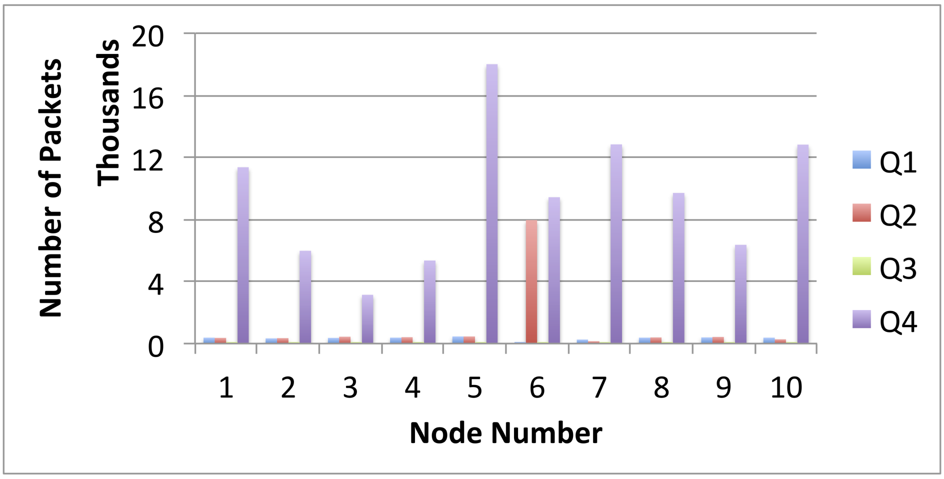


Figure 5.6: Average lengths of the queues over twelve months.

packets) during the simulation. The large difference in the number of transmitted packets (42.45% more packets transmitted by node 1) is due to the fact that node 1's queues are rarely empty while node 6's queues Q1 and Q3 are empty for extended periods of time. When the queues are empty, it means that the node uploaded all packets from these queues and has nothing to transmit on the corresponding channels until new data is gathered. This is clearly the effect of the differences in the movement of the mobile nodes in the simulations.

This simulation benefited from the availability of data collected from the deployment on wild horses, such as the GPS power consumption in terms of the time it actually took to acquire position fixes, and the horses' precise movement behaviour. Table 5.4 compares these results to the ones presented in Section 3.2.4 of Chapter 3, belonging to the same simulation scenario but without having real data input.

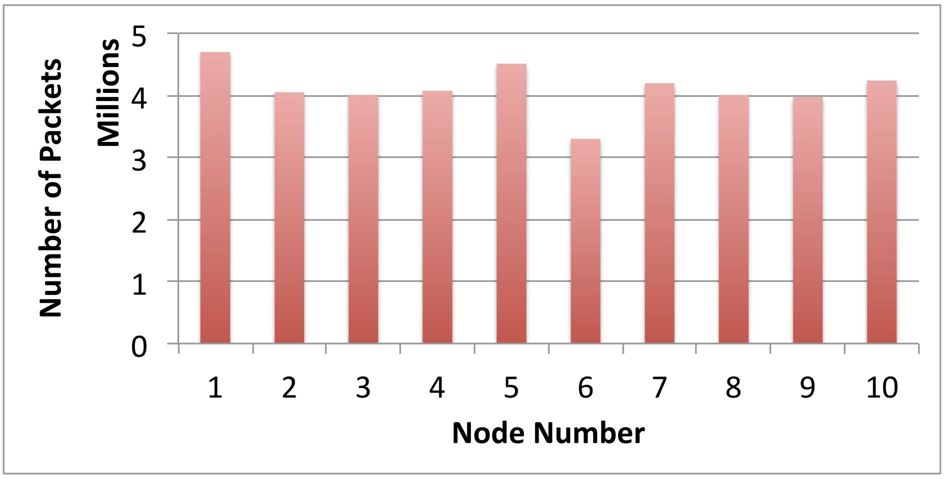


Figure 5.7: Total number of transmitted packets by each node.

In the case of the simulations with no real data input, since it was difficult to predict (prior to testing) the performance of the GPS (in terms of Time-To-Fix) in the nature reserve where the nodes were to be deployed, it was assumed that it will have a typical Warm start every approximately twenty minutes when acquiring its position. This was clearly not the case. The real Time-To-Fix would have been somewhere between a Hot start (typically one second) and a Warm start (typically thirty-three seconds), as the GPS's satellite data would be only twenty minutes old. But in the absence of any accurate information regarding the Time-To-Fix in the Doñana reserve, the value of the warm start was chosen, so as to exclude the possibility of overestimating the deployment lifetime.

As can be seen in Table 5.4, the input of real data can make a big difference in the network behaviour results. Firstly, the deployment lifetime (which in this case is represented by the lifetime of the mobile nodes) is affected by the GPS's power consumption. The lifetime of the mobile nodes being longer by 87 days (28%), is the effect of introducing the Time-To-Fix data from the GPSs used in the actual deployment, which offered a realistic simulation of the main energy consuming component. Secondly, the data collection process is mainly affected by the mobile nodes' movements, but also by the lifetime of the deployment. The latency dropping from almost three days to 2.5 hours, is the effect of introducing the actual movement of the horses in the wild as opposed to having random movement. On the other hand, the number of unique packets uploaded increasing by 26%, is a result of both the movement of the mobile nodes but also the deployment lasting for the full twelve months.

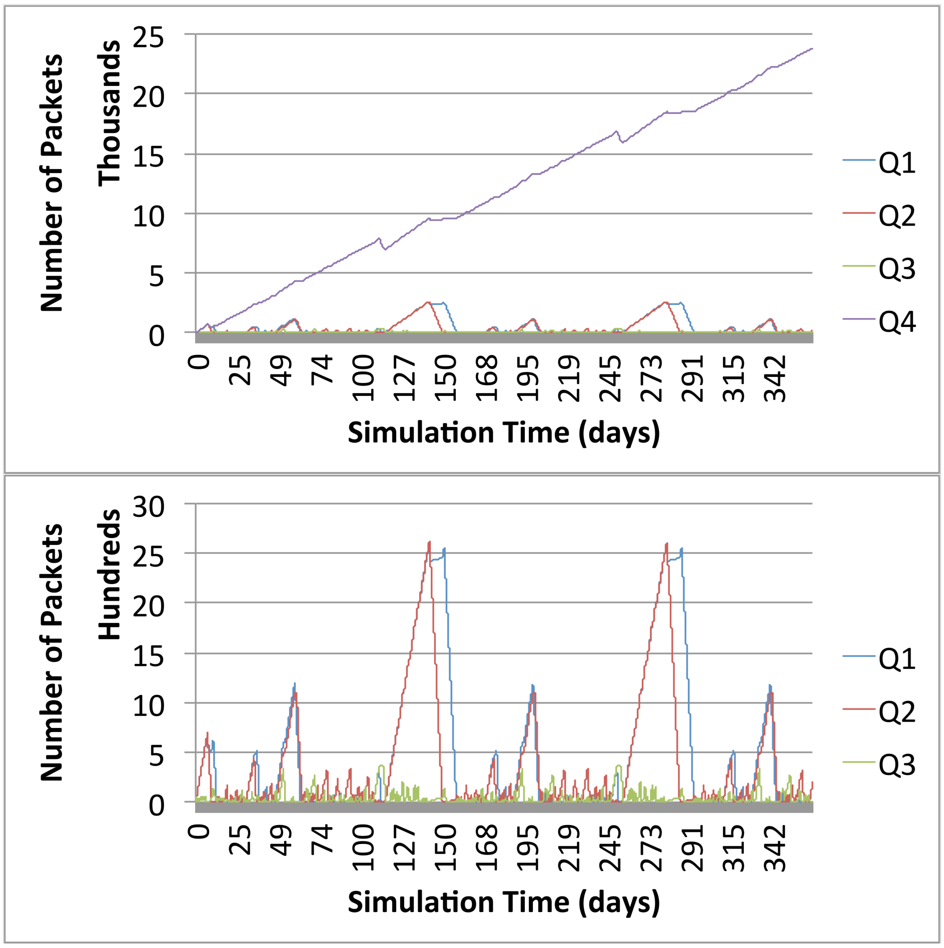


Figure 5.8: Node 1's queues: all queues (top), queues 1,2 and 3 (bottom).

Table 5.4: Comparison of Simulations With and Without Real Data Input

Metric	No Data Input	Real Data Input
Deployment length	10.18 months	12 months
Mobile nodes maximum lifetime	310 days	397 days (32 extra days - 8.24% battery left)
Average network latency	2 days, 23:27:01	02:30:56
Total number of unique packets collected	184822 (85.5%)	233844 (88.98%)
Total number of packets collected	732764	835636
Redundancy 2-way	99.85%	99.87%
Redundancy 3-way	99.21%	91.83%
Redundancy 4-way	97.29%	26.25%

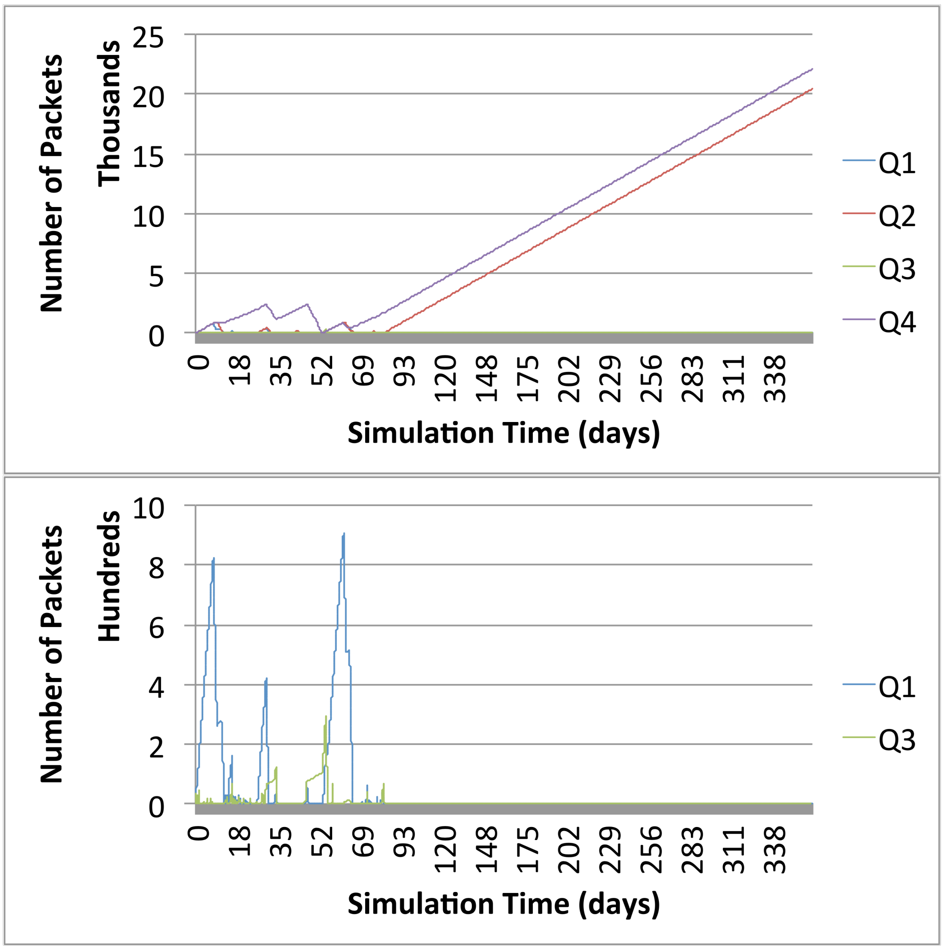


Figure 5.9: Node 6's queues: all queues (top), queues 1 and 3 (bottom).

5.1.3.2 Best- and Worst-Case Scenarios

These scenarios were simulated to better understand how the number of radio transmissions affects the power consumption, and the results are summarised in Table 5.5. The main factor that influences the number of radio transmissions in this case is the movement of the mobile nodes.

Table 5.5: Comparison of Worst and Best Cases

Metric	Number of transmitted packets	Number of delivered packets	Battery level after 12 months
Worst case	84,095,856	0	6.83%
Worst case in theory	84,096,000	0	-
Best case	1,051,200	1,051,200	12.57%
Best case in theory	1,051,200	1,051,200	-

The worst-case scenario for the battery consumption of the mobile nodes is when there is no break in uploading a packet every fifteen seconds on each of the channels. This scenario can occur if a mobile node is never in reach of a base-station, or even if it is, it never uploads all the packets it has stored from any of the queues. This is simulated by keeping all base-stations turned off during the deployment. The average battery level for the mobile nodes at the end of this deployment is 6.83%, and the total number of packets transmitted by all nodes in the network is 84,095,856, as shown in Table 5.5. This is slightly lower than the theoretical number, which does not take into account the nodes' backoff time at the beginning of the deployment and the time spent before getting the first GPS position.

The best-case scenario is when all the nodes are within range of a base-station when they are uploading a packet and no collisions occur. In order to simulate this scenario, all the mobile nodes were made static and positioned in the ranges of four base-stations, programmed one on each channel. The average battery level for the mobile nodes at the end of this deployment is 12.57%, and the total number of transmitted packets is 1,051,200, which matches the theoretical number.

The worst- and best-case scenarios show that the differences in power consumption influenced by the number of radio transmissions can make a difference of twenty days (5.5%) to the length of the deployment. This result is specific to the asynchronous data upload protocol, which requires mains-powered base-stations. This protocol showed a good performance for this scenario considering its simplicity, the amount of data uploaded, the average latency and the deployment lifetime.

5.2 VB-TDMA Experimental Validation

Validating protocols implemented in simulation environments is necessary to ensure that their behaviour and performance reflect the reality in the proposed deployment. The implementation of the VB-TDMA protocol in the SpeckSim simulator was validated against a real test deployment of eight mobile nodes having an accelerated sensor sampling rate. The test deployment was performed on the teaching herd of domestic horses at the School of Veterinary Studies in the University of Edinburgh.

The validation compared the lifetime of a mobile node from the real deployment to the lifetime of a simulated node, having the same protocol configuration and using the corresponding hardware models. Considering that the hardware models of the Prospeckz-5 platform were validated in terms of network behaviour against results from the Doñana wildlife deployment in Sections 5.1.1 and 5.1.2, by counting the number of GPS positions gathered in the node's lifetime (until the battery is depleted), we are now validating the VB-TDMA's implementation in the SpeckSim simulator. The number of GPS positions gathered is a relevant metric as the GPS module is the major contributor towards the power consumption, accounting for more than 90%.

The validation scenario consists of sampling the GPS module along with the other sensors (accelerometer, magnetometer and light sensor) once per minute, when the clock strikes the zero second. The data is uploaded every minute starting with the thirtieth second. The algorithm, besides having accelerated sampling and uploading rates and using only one channel (one base-station was deployed), it works as presented in Section 3.1.2, having 10ms Discovery slots and 100ms Upload slots.

A mobile node gathered 23,981 packets (GPS locations) in the real world before it ran out of battery, with the corresponding figure in simulation being 23,972. The validation test is considered successful, as the difference between the two experiments is extremely low: 0.03%.

5.3 VB-TDMA Simulation Results

The results presented in this section are from twelve-month simulated deployments consisting of ten horses and eight base-stations while using different MAC protocols. The simulations aim to replicate the conditions of the wildlife deployment performed in the Doñana National Park, Spain. For each simulated scenario, the mobile nodes sample the GPS module, along with the sensor module (which represents the accelerometer and the light sensor) every twenty minutes, and initiate discovery phases every five minutes (with the exception of the Asynchronous protocol which uploads a packet every fifteen seconds). Data collected from the actual deployment from ten horses over a period of forty-four days was fed into the mobility model and the GPS model used in the simulations. This replicates the movement of the horses (using the Data-Mirror mobility model) and the time it takes the GPS to acquire a position fix for every location sensed, in order to accurately estimate the wireless network behaviour and the power consumption based on real data. With the exception of the deployment running the asynchronous protocol, which had base-stations with mains power having the radio in listening mode continuously, the other deployments used battery-powered base-stations, running on three Lithium Thionyl Chloride batteries (3x2500mAh).

The metrics in the evaluation of the communication protocols were chosen for their relevance and importance to the considered class of applications. Table 5.6 presents the metrics in order of their importance, starting with the most important one: energy efficiency. As the mobile nodes deployed have limited power supply (batteries), the power consumption determines the lifetime of the deployment, and one of the most stringent requirements is having the deployments last for extended periods of time, in order to capture the effect of seasonal variations. Ideally they would last for a minimum of twelve months.

Even though mobile nodes also store the collected data in their flash memory for retrieval upon collection, it is important for the applications to provide access to the data during the deployment. This led to the need of wirelessly uploading the data to base-stations, justifying the selection of the next metric in terms of importance, the volume of data uploaded, measured as the number of unique packets uploaded to base-stations during the time of the deployment. The total number of packets that are created by ten mobile nodes in twelve months is 262,790, and is used to calculate the percentage of the uploaded data.

The third metric is latency, considered as the time between a packet's creation and its first upload to a base-station. The proposed class of applications is not too sensitive to this metric, as the latency is not an issue if it is within acceptable limits, which are specific to each application.

The last metric is network redundancy. The network redundancy is for backup purposes, and having low level of redundancy does not necessarily mean that the number of unique packets collected is low. The redundancy is inferred by the four-channel structure chosen in the example scenario of tracking and monitoring horses in the wild. This is not a direct property of any one communication protocol, however it does reflect the communication protocol's ability to upload higher volumes of packets. For other scenarios belonging to this class of applications, having this type of redundancy may not be appropriate.

5.3.1 Comparison of VB-TDMA to Asynchronous

This section compares the performance of the VB-TDMA protocol against the basic asynchronous protocol presented in Section 3.1.1. In the case of the deployments presented in this section, for both protocols, the nodes are programmed to have up to fifteen retransmissions per packet, in case an acknowledgement is not received for it. After having fifteen unsuccessful retransmissions the nodes go to sleep and wake up

Table 5.6: The Metrics Chosen for Evaluation

No	Metric	Description
1	Energy efficiency	-
2	Volume of data uploaded wirelessly	Total number of unique packets uploaded
3	Latency	Refers to the time between the creation of a packet by a mobile node and its first upload to one of the base-stations
4	Redundancy	Considers the number of packets recevied 2, 3 and 4 times, due to the use of the four different radio channels

for the next upload slot to try to upload the same packet (packets are never discarded). The time between adjacent discoveries for the VB-TDMA is five minutes and the time between uploads for the asynchronous protocol is fifteen seconds. The results are presented in Table 5.7.

The purpose of this comparison is to show that when using mains-powered base-stations, even a basic design of a data upload protocol can perform well.

5.3.2 VB-TDMA Sensitivity Analysis

The sensitivity analysis helps setting the protocol's parameters for maximising its performance for the chosen test scenario. The two parameters at the core of the VB-TDMA protocol are the time between adjacent Discovery phases and the number of retransmissions in case a packet is not acknowledged. The results for varying each of the parameters while keeping the other one constant are presented in Tables 5.8 and 5.9. Based on these results, the best trade-off between power consumption and performance is achieved with five minutes between discoveries and five retransmissions.

Table 5.7: Comparison of VB-TDMA to Asynchronous: Twelve-Month Deployment Results

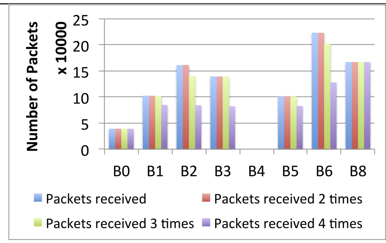
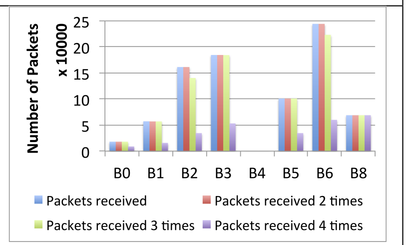
No	Metric	VB-TDMA	Asynchronous
1	Average percentage of battery left	Mobile nodes: 10.62% - 43days and 9 hours left	Mobile nodes: 8.24% - 32days and 18 hours left
2	Total number of uploaded packets	933995	835636
3	Total number of unique packets (Percentage of the received packets out of the number of expected packets)	262790 (100%)	233844 (88.98%)
4	Average network latency (hh:mm:ss)	02:28:27	02:30:56
5	Packet distribution over BSs		
6	Redundancy percentage 2-level	99.89%	99.87%
7	Redundancy percentage 3-level	91.91%	91.83%
8	Redundancy percentage 4-level	63.60%	26.25%

Table 5.8: VB-TDMA Sensitivity - Time Between Discoveries (Number of Retransmissions = 15)

No	**Time Between Discoveries**	**1 min**	**5 min**	**10 min**	**25 min**	**75 min**	**125 min**
1	Average percentage of battery left	Nodes: 10.02%; BSs: 57.4%	Nodes: 10.62%; BSs: 61.82%	Nodes: 10.70%; BSs: 62.37%	Nodes: 10.74%; BSs: 62.70%	Nodes: 10.764%; BSs: 63.04%	Nodes: 10.768%; BSs: 63.15%
2	Total number of uploaded packets	934002	933995	933989	928180	922487	922370
3	Total number of unique packets (Percentage of the received packets out of the number of expected packets)	262790 (100%)	262790 (100%)	262790 (100%)	262790 (100%)	262782 (99.99%)	262734 (99.97%)
4	Average network latency (hh:mm:ss)	02:24:01	02:28:27	02:34:26	02:41:54	03:12:19	03:44:17
5	Redundancy percentage 2-level	99.900%	99.899%	99.899%	99.877%	99.877%	99.883%
6	Redundancy percentage 3-level	91.913%	91.912%	91.911%	91.836%	91.809%	91.818%
7	Redundancy percentage 4-level	63.603%	63.602%	63.601%	61.487%	59.359%	59.363%

Table 5.9: VB-TDMA Sensitivity - Number of Retransmissions (Time Between Discoveries = 5 minutes)

No	**Number of Retransmissions**	**1**	**5**	**10**	**15**
1	Average percentage of battery left	Nodes: 10.77%; BSs: 63.15%	Nodes: 10.71%; BSs: 61.82%	Nodes: 10.67%; BSs: 61.82%	Nodes: 10.62%; BSs: 61.82%
2	Total number of uploaded packets	922370	933995	933995	933995
3	Total number of unique packets (Percentage of the received packets out of the number of expected packets)	262734 (99.97%)	262790 (100%)	262790 (100%)	262790 (100%)
4	Average network latency (hh:mm:ss)	03:44:17	02:28:27	02:28:27	02:28:27
5	Redundancy percentage 2-level	99.883%	99.899%	99.899%	99.899%
6	Redundancy percentage 3-level	91.818%	91.912%	91.912%	91.912%
7	Redundancy percentage 4-level	59.363%	63.602%	63.602%	63.602%

5.3.3 Comparison of VB-TDMA to Other MACs

As presented in Section 2.2.2, most of the recent MAC protocols have changed their emphasis from energy efficiency to other metrics, such as throughput and delay [89, 86, 87, 88]. Since the class of applications targeted in this thesis consists of long-term deployments of sparse (low node density) sensor networks with a low level of traffic, existing protocols which are not tailored for this type of applications perform poorly. This can be seen in Table 5.10 and Figure 5.10. The MAC protocols chosen for comparison were selected from those that would have the best chances to perform well for this type of applications, thus belonging to the category of MACs that have energy efficiency as their primary concern. However, since the proposed VB-TDMA protocol is specifically designed for this class of applications, it significantly outperforms the rest.

The chosen protocols for comparison are S-MAC [70] a WSNs MAC protocol aiming to reduce energy consumption and supporting self-configuration, SpeckMAC-D [69] a low-power distributed, unsynchronised, random-access MAC protocol for mobile ad-hoc WSNs, and XMAC [75] a WSNs MAC protocol that performs low power listening by employing a shortened preamble approach. The well-known B-MAC [73] was not selected as it has been proven both theoretically and experimentally that it is outperformed by SpeckMAC-D in terms of energy efficiency [69]. Dozer [118], a low-power protocol designed for periodic data collection combining the MAC and routing layers, was also not selected. The protocol builds a tree-based network structure, routing packets towards the root, which makes it unsuitable for highly mobile networks.

CarrierSenseMac is not considered an energy efficient MAC, but it was selected to highlight a worst-case scenario of having the radio turned on continuously.

The MACs selected for comparison are representative of the class of low-power MACs, covering both the scheduled (SMAC - employs time slots in a fashion similar to TDMA) and random-access (XMAC and SpeckMAC-D) categories. Their configurations, while aiming for maximising the deployment lifetime, consider the trade-off between the nodes' lifetime and the amount of data uploaded.

The results show that after twelve months the mobile nodes running the VB-TDMA protocol had more than 10% of their battery remaining, which would enable them to run for another 43 days. In comparison, all the other MAC protocols completely discharged the mobile nodes' batteries before the end of the deployment. Also, the number of packets (unique or including duplicates) collected by base-stations with the VB-TDMA is significantly higher than for the rest of the protocols. Although XMAC collected the next highest number of packets, it still represents less than 50% of the number of unique packets collected by VB-TDMA.

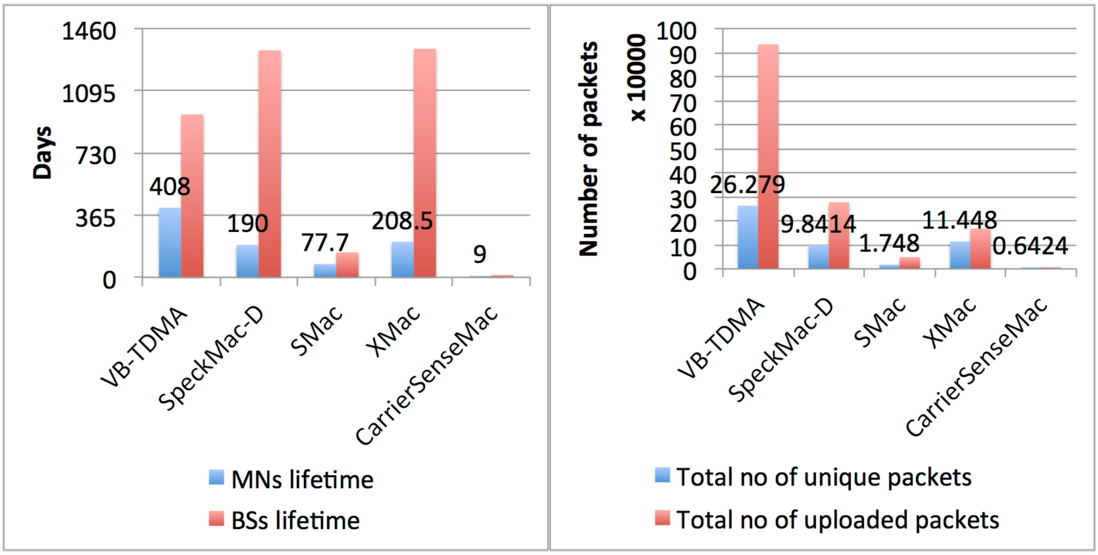


Figure 5.10: The lifetime of the nodes (left); The number of packets collected during a twelve-month deployment (right).

Table 5.10: Comparison of VB-TDMA to Other MACs: Twelve-Month Deployment Results

No	Metric	VB-TDMA	SpeckMac-D	SMac	XMac	CarrierSenseMac
1	Average percentage of battery left	Mobile nodes: 10.71% - 43days and 18 hours left; Base-stations: 61.82%	Mobile nodes: 0% (depleted after 190 days); Base-stations: 72.6%	Mobile nodes: 0% (depleted after 77.7 days); Base-stations: 0% (depleted after 146 days)	Mobile nodes: 0% (depleted after 208.5 days); Base-stations: 72.79%	Mobile nodes: 0% (depleted after 9 days); Base-stations: 0% (depleted after 13.6 days)
2	Total number of uploaded packets	933995	277061 (29.66% of VB-TDMA)	48953 (5.24% of VB-TDMA)	168262 (18.01% of VB-TDMA)	6841 (0.73% of VB-TDMA)
3	Total number of unique packets (Percentage of the received packets out of the number of expected packets)	262790 (100%)	98414 (37.44%)	17480 (6.65%)	114480 (43.5%)	6424 (2.44%)
4	Average network latency (hh:mm:ss)	02:28:27	02:22:43	18:19:47	12:24:40	02:04:49
5	Redundancy percentage 2-level	99.89%	89.47%	99.72%	33.70%	3.25%
6	Redundancy percentage 3-level	91.91%	87.39%	80.31%	13.27%	3.23%
7	Redundancy percentage 4-level	63.60%	4.66%	0.01%	2.00%	0%

5.3.4 VB-TDMA Scalability

The protocol's scalability reflects the capability of the solution to handle a growing number of mobile nodes. This section focuses on the network's behaviour and performance when increasing the number of mobile nodes in the deployment. The protocol has a clear limitation, as the base-stations' power consumption strictly depends on the number of mobile nodes in the network. The experiments also looked at the network latency, the number of packets uploaded and the memory usage of the base-stations.

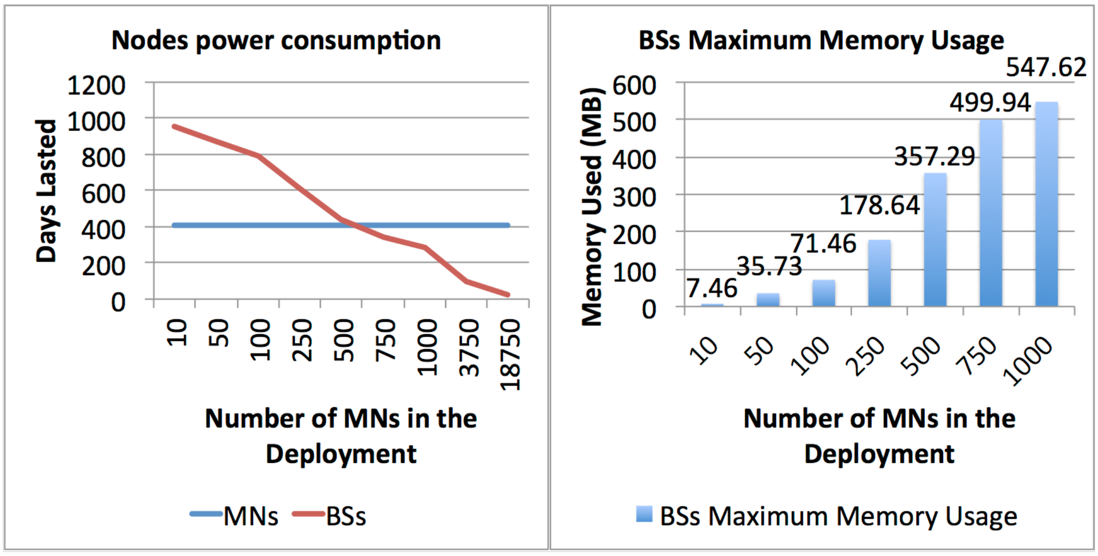


Figure 5.11: Nodes' power consumption (left); Base-station maximum memory usage (right).

While the number of mobile nodes in the deployment does not significantly affect their own power consumption, the base-stations' power consumption increases linearly with the number of mobile nodes. This can be viewed in the left-hand side of Figure 5.11, which shows a clear decrease of the base-stations' battery lifetime with the increase of the number of mobile nodes in the deployment. The results presented in this figure are for base-stations powered by three Lithium Thionyl Chloride batteries (3x2500mAh). The base-stations do not use solar cells by default, as the intention is to keep the solution self-contained, eliminating dependencies on external factors that may limit its applicability in certain scenarios (e.g., deployments in areas with low or no light, deployments in dusty areas, or scenarios where it is unlikely that the transparent lid of the enclosure will remain clean during the deployment). The use of solar cells would also require using rechargeable Lithium Polymer batteries, which bring other limitations, such as not being suitable to operate in extreme temperature condi-

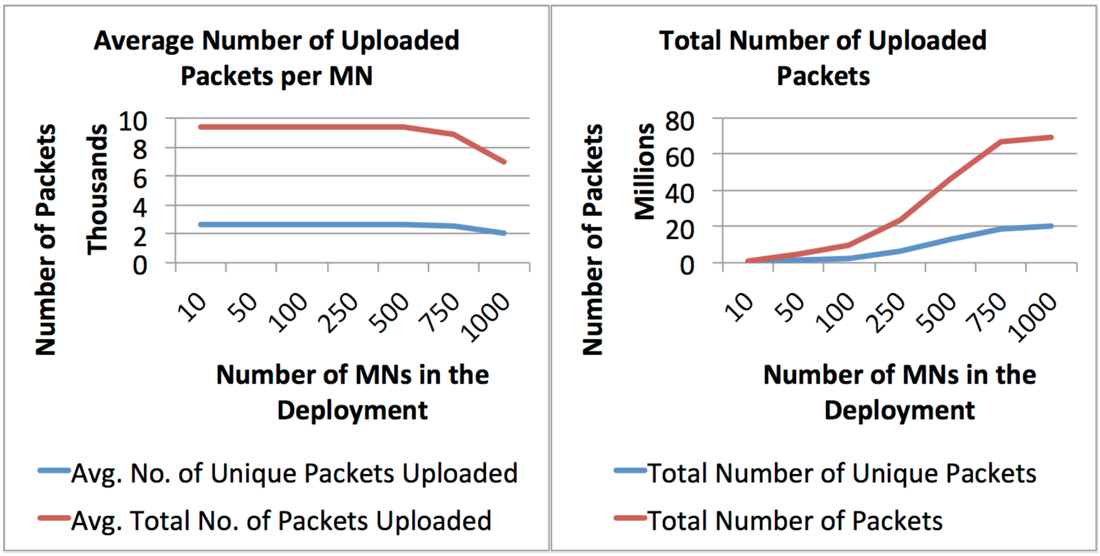


Figure 5.12: Average number of uploaded packets per mobile node (left); Total number of uploaded packets (right).

tions.

The right-hand side of Figure 5.11 shows that the initial design of the base-stations relying on storing data only on the Prospeckz's built-in 8MB flash memory would be adequate for twelve-month deployments of up to ten mobile nodes. For larger deployments where base-stations require more storage space, the simplest solutions are either using external storage options such as SD cards, or further uploading the collected data to a collection point with higher storage capabilities. The deployment on wild horses in Spain had the base-stations store the data on SD cards without any power consumption concerns, as they were mains powered. The graph does not show a completely linear increase of the base-stations' maximum memory usage, as the results represent the amount of data actually collected by the base-stations in simulations. The reason for this is the VB-TDMA's differences in its TDMA schemes when the number of mobile nodes increases. In the case of the 750 and 1000 node deployments, the base-stations collected less data than would be expected because the mobile nodes ran out of battery before the end of the twelve-month deployments.

Figure 5.12 shows that even though more packets are collected by the base-stations when having more mobile nodes in the deployment (right-hand side), the average number of packets collected from one mobile node decreases (left-hand side). This justifies the less abrupt increase in the number of collected packets from 750 to 1000 mobile nodes that can be noticed on the right-hand side of this figure. This behaviour is due to the base-stations' battery lifetime being shorter than twelve months (284 days) in the case of the deployment with one thousand mobile nodes.

The latency variation is dependent on more factors than the number of mobile nodes, such as the chosen scenario, the movement model and the deployment lifetime. The left-hand side of Figure 5.13 shows that in our case the latency slightly increased when the number of mobile nodes in the deployment increased beyond five hundred. As for the redundancy levels presented in the right-hand side of this figure, they do not show significant changes with the variation of the number of mobile nodes. The levels dropped slightly in the case of the 1000-mobile-node deployment due to the base-stations batteries depleting before the deployment reached twelve months.

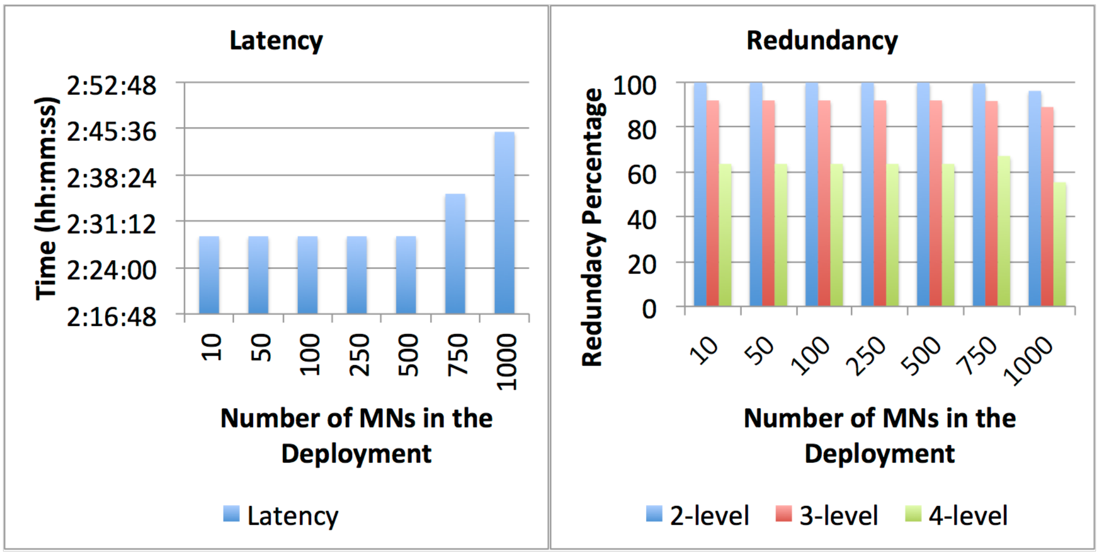


Figure 5.13: Latency (left); Redundancy (right).

The results presented in this section show that the protocol's ability to scale in terms of the number of mobile nodes in the deployment is highly dependent on the base-stations' resources: firstly the amount of power available, and secondly their storage capabilities. Using the current deployment infrastructure, the protocol is able to scale to up to seven hundred nodes. Another limiting factor for its scalability is the time between discoveries. For example, if it is desired to maintain the five-minute interval between Discovery phases, this limits the Discovery and Upload phases to five minutes, which in turn limit the deployment size. If we assume that only 10% of the mobile nodes are in range with one base-station at a time, and that it would be satisfactory to have only one Upload slot per node after each Discovery phase, this limits the deployment size to 15,000 mobile nodes.

5.3.5 The Multihop (Store-and-Forward) Functionality

Enabling communication and data exchange between mobile nodes can lead to lowering the latency in the data upload process and increasing the amount of data collected. Depending on the application scenario, the benefits of this feature may outweigh its power consumption implications. This section presents the results of the VB-TDMA algorithm with the multihop functionality enabled. The multihop functionality (described in more detail in Section 3.1.2.2) enables the exchange of data packets among mobile nodes. It aims to solve the data uploading problem of mobile nodes that do not come in contact with a base-station for extended periods of time, but during these periods come in contact with other mobile nodes that visit the base-stations more frequently. At the core of the multihop functionality is a threshold for the time of the last upload to a base-station, which determines whether a mobile node uploads data packets to other mobile nodes. Selecting an appropriate threshold is scenario-dependent. Mobile nodes do not upload the same data packet to more than one mobile node, and they do not upload data collected from mobile nodes to other mobile nodes. This limits the mobile node network to carrying only one copy of a mobile node's data packets. When a mobile node comes within range of a base-station, it uploads all its data on that channel, followed by the data collected from other mobile nodes.

In the case of the scenario of tracking wild horses, the multihop feature with the sending/receiving threshold set to twenty-four hours and the time between mobile-node discoveries set to five minutes, lead to a 27% decrease in the latency. This result comes at a fairly insignificant price of a 1.81% increase in the mobile nodes' power consumption, while the number of unique packets collected by the network remains unchanged (see Table 5.11). In the case of the "cyclists" application presented in Section 2.1.2, which crowd sources air quality data from cyclists who use bicycles equipped with air quality sensors, enabling the multihop feature had a greater effect on the packet collection latency. In this scenario the daily movements of ten student cyclists at the University of Edinburgh were simulated, with data being uploaded to three base-stations. For this application, the multihop feature lowered the data packet latency by more than 50% with only a small increase in the power consumption (see Table 5.12). Even though the effects of the multihop functionality to reduce the network latency are significant, with only a slight increase in the mobile nodes' power consumption, this feature could have an even greater impact on the network performance (in terms of the network's latency and amount of data collected) in scenarios where some of the mobile nodes are rarely or never in range of a base-station.

We used the platform and data-mirror mobility models to compare the network performance of the VB-TDMA Multihop to the network performance obtained with the MAC protocols selected in Section 5.3.3, but having the store-and-forward multihop functionality added. An alternative approach would have been to compare the performance of the VB-TDMA Multihop protocol to that of a routing protocol running over the other MACs. However, this would have been inappropriate for the long-term tracking and monitoring of horses in the wild and similar scenarios, as a routing protocol would have offered a lower performance in the data upload process. When using a routing protocol, to upload data from a mobile node out of reach of a base-station, a path of mobile nodes to a base-station must exist and be maintained during the upload. This is unlikely due to the limited communication ranges between mobile nodes in relation to the large size of the area inhabited by the horses, and the fact that they travel predominantly in groups (as shown in Figure 4.13 in the previous chapter).

The comparison results from Table 5.13 show that only the mobile nodes running the VB-TDMA lasted for the entire twelve-month duration of the deployment, with 36.5 days worth of battery life to spare. This is a reason why the network running the VB-TDMA collected more packets compared to the other protocols. The next best performance was that of the nodes running XMAC, which lasted for 204 days, but uploaded to the base-stations less than 50% of the data that would have been collected over twelve months. At the end of the deployment, the base-stations running SpeckMac-D and XMAC were left with approximately 17% more battery than the ones running the VB-TDMA. Although this has no impact on the simulated deployment, it may prove to be meaningful if this percentage persists when scaling up the number of mobile nodes.

Table 5.11: VB-TDMA Multihop: Twelve-Month Wildlife Deployment

No	Metric	VB-TDMA	VB-TDMA Multihop
1	Average percentage of battery left	Mobile nodes: 10.71%; Base-stations: 61.82%	Mobile nodes: 9.09%; Base-stations: 61.81%
2	Total number of unique packets (Percentage of the received packets out of the number of expected packets)	262790 (100%)	262790 (100%)
3	Average network latency (hh:mm:ss)	02:28:27	01:47:15 (27.75% smaller)
4	Redundancy percentage 2-level	99.89%	99.89%
5	Redundancy percentage 3-level	91.91%	91.63%
6	Redundancy percentage 4-level	63.60%	2.62%

Table 5.12: VB-TDMA Multihop: 24-Hour Student Cyclists Deployment

No	Metric	VB-TDMA	VB-TDMA Multihop
1	Average percentage of battery left	Mobile nodes: 95.63%; Base-stations: 98.23%	Mobile nodes: 95.60%; Base-stations: 98.23%
2	Total number of uploaded packets	14036	14036
3	Average network latency (hh:mm:ss)	00:47:03	0:19:36 (58.34% smaller)

Table 5.13: Comparison of VB-TDMA to Other MACs (Multihop Enabled): Twelve-Month Deployment Results

No	Metric	VB-TDMA	SpeckMac-D	SMac	XMac	CarrierSenseMac
1	Average percentage of battery left	Mobile nodes: 9.09% - 36.5 days left; Base-stations: 61.81%	Mobile nodes: 0% (depleted after 189 days); Base-stations: 72.5%	Mobile nodes: 0% (depleted after 73 days); Base-stations: 0% (depleted after 148 days)	Mobile nodes: 0% (depleted after 204 days); Base-stations: 72.2%	Mobile nodes: 0% (depleted after 8.9 days); Base-stations: 0% (depleted after 13.8 days)
2	Total number of unique packets (Percentage of the received packets out of the number of expected packets in 12 months)	262790 (100%)	113610 (43.23%)	16596 (6.31%)	125513 (47.76%)	6388 (2.43%)
3	Average network latency (hh:mm:ss)	01:47:15	02:20:10	16:03:28	12:02:01	02:06:23

The lower latency of the VB-TDMA Multihop compared to the other MACs, could be explained by the fact that the VB-TDMA is collision free, as each node has its own time slot for sending data, and the synchronisation process of the nodes does not impose any additional load on the radio communication channels. However, since for this scenario the latency is mainly affected by the movement of the mobile entities, and the VB-TDMA mobile nodes lasted significantly longer than the nodes running the other MACs (between 161 to 292 days), it should also be considered that the movement of the mobile nodes in the final third of the deployment may have generated a lower latency.

5.4 Summary

This chapter described the validation of the mobility and hardware models implemented in the SpeckSim simulator, which were used to analyse the performance of the VB-TDMA protocol, and to compare it to a selection of low-power MACs in the context of tracking and monitoring mobile entities in the outdoors. As the VB-TDMA was specifically designed for this class of applications, it performed significantly better, offering an almost twice as long deployment lifetime and collecting more than double the number of unique packets than the MAC protocol with the next best performance. The scalability study of the protocol revealed three principal limiting factors: the battery capacity available to the base-stations, the data storage capabilities of the base-stations, and the time interval between Discovery phases. The results are promising, showing that with the current infrastructure setup, the protocol can scale up to seven hundred nodes.

The next chapter reaffirms the thesis statement, discusses the issues, and reaches a final judgement. This is followed by suggestions of future work and improvements that could further lengthen the deployment lifetime and generally improve the performance of a network running the VB-TDMA protocol.

Chapter 6

Conclusions and Future Work

6.1 Summary and Conclusions

For the past two decades, considerable advances in embedded hardware technology, from microcontrollers to sensors, have led to significant reduction in sizes and energy consumption, and increases in performances in terms of computational power, sampling rates and throughput. Embedded sensing platforms have shrunk to the size of coins, while still providing considerable computational power (e.g., 32-bit processors at up to 48MHz [106]), along with wireless and sensing capabilities. This enabled the development of new applications in domains such as wearable computing and the Internet of Things.

In the case of applications involving mobile sensing, tagging the sensor data only with temporal information is not adequate; these types of applications and scenarios require more contextual information, such as the locations of the sensor readings. Applications of long-term tracking and monitoring of mobile entities in the outdoors come with considerable challenges, requiring a high degree of functionality over extended periods of time, and all on a highly constrained energy budget.

Solutions which target the entire class of applications do not seem to exist, only ones targeted at specific scenarios part of this class. The most widely used approach, and the one adopted in this thesis, is to attach GPS- and wireless-enabled devices to the mobile entities that require tracking. Although the literature presents several other options, many in the context of wildlife tracking, e.g., infrastructure of sound activated video cameras [22], RFID tags [19], satellite imaging [119, 120], the proposed approach offers superior performance in terms of the combination of infrastructure setup, positional accuracy, cost, deployment lifetime, and the ability to monitor handling of objects or behaviour of animate beings.

The main challenges are in the long-term operation of devices limited in size and weight, constantly using power-hungry components such as the GPS and radio. The processes of locating the nodes and wirelessly uploading the data are by far the most power-hungry ones (e.g., together they represent over 98% of the overall power consumption in the case of the horse tracking application), making the rest of the system's functionalities almost insignificant in terms of their energy consumption. Since applications within this class require accurate locations at a certain frequency, reducing the sampling rate of the GPS, or avoiding it all together is not an option. Therefore the approach has been to minimise the other major source of energy consumption, the wireless data upload.

The proposed solution for this class of applications is centred on the novel Virtual Beacon TDMA data upload protocol, which minimises the power consumption for wirelessly accessing the data on the mobile nodes. It achieves this by eliminating the major sources of energy wastage characteristic of MAC-layer communication protocols such as collisions, overhearing, control packet overhead and idle listening. This is possible by exploiting the GPS's ability to provide accurate time, which for this class of applications comes at no extra energy cost. The GPS time is used to obtain a millisecond-accurate synchronisation of the nodes, free of any communication overhead. The communication is facilitated by a TDMA scheme that uses both statically and dynamically allocated time slots for collision-free communication between the nodes. The protocol is paired with a new low-power hardware platform that benefits from the latest generation components, together offering a high performance (in terms of the amount of data uploaded) to energy consumption ratio.

The solution was validated and tested in both simulations and real deployments for the specific scenario of long-term tracking and monitoring wild horses, which is representative of the targeted class of applications. Performing a deployment on the Retuerta, a rare breed of Spanish wild horses, was a unique opportunity. The endangered status of these horses and the challenges of long-term wildlife tracking motivated the use of a WSN-based solution. The requirements in terms of the deployment lifetime, the frequency and accuracy of GPS locations and the restrictions on the size and weight of the mobile devices, made existing mobile WSNs wildlife tracking systems unsuitable for this scenario. According to the performance evaluation results, the proposed solution outperforms existing off-the-shelf animal tracking systems in terms of deployment lifetime. The mobile nodes built with the Prospeckz-5 platform are several times smaller and lighter than the existing state-of-the-art solutions, without compromising on deployment lifetime or the frequency and accuracy of GPS positions. This is possible by eliminating the energy consumption overhead of synchronising the communication of the nodes for uploading the sensor data. Although favourable circumstances such as having mains-powered base-stations may allow the use of basic asynchronous upload methods that can offer good performance in the data collection process, these would not be viable solutions for scenarios using battery-powered base-stations. Overcoming this challenge motivated the design of the VB-TDMA.

The VB-TDMA protocol was successfully implemented and tested, and performed significantly better in terms of the battery lifetime than a selection of low-power MAC protocols considered possible alternatives for this scenario. The low population densities and the irregular movement patterns of the horses make most of the existing MAC protocols inefficient in terms of energy conservation. The superior performance of the VB-TDMA protocol was validated in accurate simulations, using validated simulation models against actual deployments, and inputs of real data on movement patterns gathered from the deployment. With configurations aiming for a balanced trade-off between energy conservation and amount of uploaded data, and with the same hardware resources, the VB-TDMA offers twice the deployment lifetime with more than double the amount of data uploaded. The protocol also offers good scalability potential (of up to 500-700 mobile nodes, using the current node configurations and infrastructure) for scenarios in this class of applications.

The two real deployments demonstrated the solution's ability to track and monitor the behaviour of animals, allowing continuous access to this data wirelessly. We have presented unique insights into the individual and group behaviour of Retuerta horses, ranging from information on daily activity levels to group behaviour such as time spent in groups, how group memberships change and evolve over time, and the correlation of the horses' movements to seasonal changes in their environment in relation to precipitation and temperature.

6.2 Discussion

Although this solution is presented as a generic one for the entire class of applications of long-term tracking and monitoring of mobile entities in the outdoors, it requires certain configurations to meet the specific requirements of different scenarios.

Data Upload Protocol. The design of the application running on the nodes is built around the wireless communication process. The proposed VB-TDMA protocol comes with several options that require careful consideration when tuning it for particular scenarios, such as:

- Setting the time between the Discovery phases, based on the desired responsiveness of the network.
- Setting the maximum number of retransmissions of a packet, based on the level of noise, interferences or obstacles found in the 2.4 GHz spectrum in the area of the deployment.
- Setting the size of the Discovery slots, which would be related to the number of retransmissions of a packet and the number of different channels used. For example, in the case of an interference-free environment, where one or maybe two retransmissions would be considered, along with the use of only one channel, the Discovery slots could be chosen somewhere in the vicinity of two milliseconds.
- Setting the size of the Upload slots. The following need to be taken into consideration when choosing an Upload slot size.
  - The radio transfer rate (in the case of the horse tracking scenario its maximum transfer rate of 2Mbps was used).
  - The size of the network, the expected mobile node densities around base-stations.
  - The volume of sensor data generated that is required to be uploaded
  - The behaviour of the mobile nodes, such as the estimated time when they are out of range of the base-stations, and the length of the contacts between mobile nodes and base-stations.
  - Fairness (e.g., when multiple nodes are in range of a base-station for a short period of time, it is considered fair that each of them would upload part of their data, rather than only one of them using up the entire time to upload its data).
  - Each Upload slot implies approximately 2ms of idle listening for the base-stations. For every upload slot the base-stations are programmed to start listening time for packets approximately one millisecond before and to continue listening one millisecond after each mobile nodes' slot. Also, during an upload slot if the base-station does not receive packets for X milliseconds (with X chosen based on the number of retransmissions of a packet), it is considered that the mobile node has either finished the upload or went out of range, thus, the base-station terminates the upload phase for that mobile

node. Programming the protocol to use a large number of shorter Upload slots increases the power consumption on the base-stations' side due to the idle listening before and after each slot. On the other hand, having very large Upload slots may affect the fairness of the data upload process (e.g., some nodes may upload a high volume of data while others may not get the chance to upload any before they are out of range of the base-station). The expected mobile node densities around base-stations, the volume of data to be uploaded, and the fairness option in the data upload process, are indirectly proportional to the size of the Upload slot; whereas the average time mobile nodes are in range of base-stations, and the desire to reduce the idle listening times for the base-stations generated by each Upload slot, are directly proportional to the Upload slot size.

- Choosing the number of different channels to use. The use of multiple communication channels eliminates the problem of packet collisions when multiple base-stations are covering the same deployment area. It is also a good way of providing redundancy in the packet collection process if desired.
- Enable/disable the multihop functionality. When enabled, the send/listen threshold has to be set, based on the expectations regarding the amount of time that the mobile nodes would be out of reach of the base-stations.

All of these parameters provide flexibility to the VB-TDMA protocol, customising it to meet the different requirements of applications within this class.

Sensor Data Collection. The sensor data collection process includes the algorithms used for processing sensor data on the nodes. In the case of sensors such as accelerometers or magnetometers, sampling rates of the order of 10Hz are required to extract useful information from the data. In the case of long-term deployments, it is unlikely that the raw data from these sensors would fit on the limited flash memory of the mobile nodes. Also, it would be impractical in terms of power consumption and data rate to upload all the raw data over the radio. These reasons justify processing the data locally on the nodes, and the values obtained reflect the resulting behaviour of the sensor data during the selected period. For example, the mobile nodes used in the Doñana deployment sampled the accelerometer continuously, while storing only two values for every twenty-minute interval: the head orientation and the maximum activity of the horses. Other applications may require different information to be extracted from the accelerometer data, or the use of other sensors. The online sensor data processing is usually customised for the needs of each application.

The Hardware Platform. The Prospeckz-5 board has proven itself during the deployment, but would benefit from replacing two components: the FastraxUC430 GPS module and the Flash chip. Its small footprint and its extensibility (allowing the attachment of external sensors), make it versatile, suitable for most applications, and although its design dates to late 2012, it is still currently at the pinnacle of low-power hardware platforms for wireless sensor networks.

The platform is highly transferable to other applications within our class. In many applications, with the exception of the ones that specifically require different node designs, the current proposed design of the mobile nodes and base-stations used for tracking the horses, are highly recommended. For example, the same mobile nodes (without the strap) were successfully used in another application for monitoring the water properties in a lake in the Doñana Park. This required the attachment of two new sensors to the Prospeckz-5 platform for measuring the acidity and dissolved oxygen levels. The sensor data was uploaded to mobile base-stations carried by UAVs flying over the lake.

Data Analysis. Handling large sets of data in a timely manner requires efficient ways for processing them. The majority of the Bash and Python scripts developed for interpreting the movement data from the real deployments and the simulations presented in this thesis, can be easily adapted to other applications within the targeted class. However, some of the scripts that make use of algorithms such as the ones designed for determining the groups of mobile nodes or detecting the lack of movement, may only be appropriate for similar applications in animal tracking.

Simulation Models. With minimal parameter configurations, the simulation models can be used to simulate any other scenario belonging to the targeted class of applications. The radio and GPS models were customised to match the behaviour of these components in the Doñana environment. These can be changed easily to reflect their behaviour in other environments if data is available, or to revert to using the parameters in the datasheets. The highly modular code structure, makes it simple to add new models of hardware components or to change existing ones, for keeping up with future design changes of the Prospeckz-5 platform. Furthermore, any specific movement of the mobile nodes can be achieved in simulations by importing the appropriate movement input files. Movement patterns can also be generated by choosing the average distance to be travelled and the average speed, along with the standard deviations of both parameters, to offer a spread of distances and speeds while controlling the values that these aim to average to.

When designing such a system it is always a good idea to keep in mind the main metric of focus, and consider it in the design of each component. For this class of applications, this metric was energy consumption. Optimizing the energy consumption was considered throughout all the design stages of the solution including both hardware and firmware.

In the case of tracking and monitoring wild horses, the hardware platform was new, custom-designed, making use of the latest low-power components (microcontroller, radio, sensors including GPS). As stated, the application running on this platform eliminated the majority of the power wastage processes that usually come with the wireless upload of data: collisions, overhearing, and control packet overhead, and idle listening was minimised. Besides optimising the wireless communication, the energy consumption was further reduced through other methods such as adjusting the GPS "ON" times for achieving the desired trade-off between energy consumption and positional accuracy, and avoiding reading data from the flash by using the RAM for the data upload process when possible.

Our experiences made it clear that successful deployments of large scale systems over extended periods of time (e.g., twelve months) require meticulous preparation, testing and forward planning which should take into account the environment and local conditions of the deployment site as much as possible. Detailed knowledge and a thorough analysis of the applications scenarios is also highly valuable, as these may lead to identifying potential optimisations.

6.3 Future Work

As demonstrated throughout this thesis, the proposed solution addresses the challenges faced by this class of applications. However, features can be added to maximise the energy efficiency, thereby further extending the potential deployment lifetime, and there is scope to add new functionalities. Some possible improvements are discussed below:

Base-stations in range: If the application does not require the use of mobile base-stations, and relies only on a network of fixed static base-stations, the number of radio transmissions can be easily reduced by having the mobile nodes use their latest GPS location for checking if it is likely to be in range of one of the base-stations. This requires the mobile nodes to be aware of the fixed locations of the base-stations, which can be hardcoded before deployment. If some of these locations are to change over the duration of the deployment, a mechanism to communicate the changes can be implemented. However, if frequent location changes are to be expected, having this feature may prove to lower the network's performance in terms of energy consumption, amount of data uploaded and latency.

Wireless reprogramming: Enabling mobile nodes to be reprogrammed wirelessly would also be a solution for the previously mentioned issue of changing the locations of the static base-stations, but more importantly, it can enable customising and adjusting the data collection process of an ongoing deployment. In the case of wildlife monitoring, it can enable ecologists to tailor the data collection to their current research. The difficulties and restrictions of recapturing wild animals, and the current communication and lifetime capabilities exhibited during deployments, motivate the benefits of this functionality. However, there are still challenges to be met with this functionality due to the limited communication range and the faulty nature of the devices in harsh environments.

Spacing out GPS locations: Depending on the application, it may be possible to reduce the GPS sampling frequency, while still providing the required number of locations for the mobile entities, by predicting one location between every two GPS ones. In the case of terrestrial animal tracking applications, since the accelerometer data can be used for indoor people tracking [35, 121, 122], it is worth researching to what extent the accelerometer and magnetometer can be used to track animals' movements. Even though a lower positional accuracy is expected compared to the GPS, if it is within the acceptable limits of the application, it can be used to space out in time the GPS sampling, and thereby increase the deployment lifetime. The method proposed (Figure 6.1) requires performing another real deployment in order to obtain a significant body of data for training and testing a machine learning algorithm. The data needed would be the following:

- The resultant direction (vector magnitude) calculated from the magnetometer data over the past sampling interval.
- Some cumulative indication of activity (e.g., average activity or number of steps) over the past sampling interval.
- The previous GPS location at time t_1.

– The succeeding GPS location at time t_3.

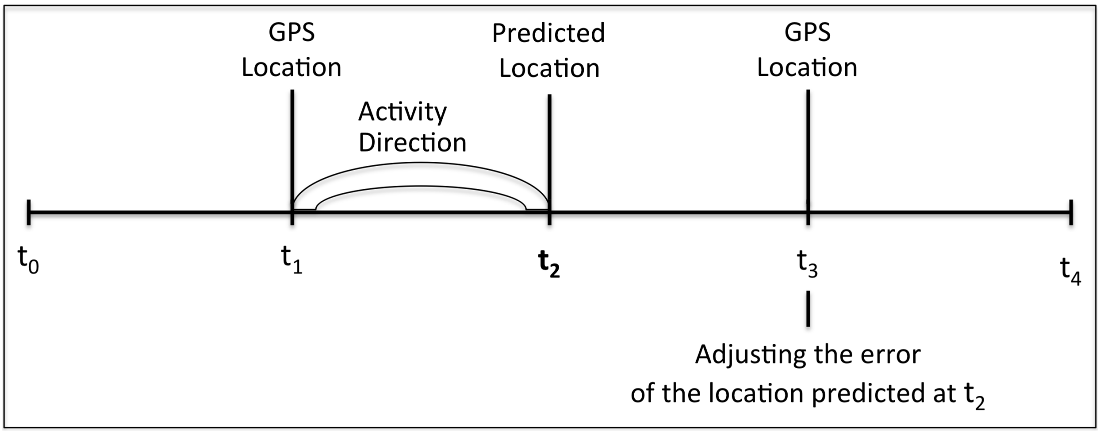


Figure 6.1: Spacing out GPS locations in time.

Starting from the latest / most recent GPS location (Lt_1), the magnetometer data gives the general direction of the mobile node's movement, and the accelerometer data is used to estimate the distance travelled by the animal. By training a linear regression machine learning algorithm on a large set of this data, it is possible to predict the current location (PLt_2) of the mobile node based on the above information. Considering the GPS location Lt_3 and assuming that the error in the predicted location accuracy grows linearly with time, some backtracking error adjustation can be performed to improve the accuracy of the location predicted at time t_2.

The VB-TDMA protocol developed as a solution for uploading data for the emerging class of problems in tracking and monitoring mobile entities has been demonstrated to fulfil the energy efficiency requirements. This protocol in combination with the Prospeckz-5 nodes offers a systemic solution for this class of applications.

Bibliography

[1] D. Puccinelli and M. Haenggi, "Wireless sensor networks: applications and challenges of ubiquitous sensing," *IEEE Circuits and systems magazine*, vol. 5, no. 3, pp. 19–31, 2005.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[3] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 88–97, ACM, 2002.

[4] T. Arampatzis, J. Lygeros, and S. Manesis, "A survey of applications of wireless sensors and wireless sensor networks," in *Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005.*, pp. 719–724, IEEE, 2005.

[5] B. Warneke, M. Last, B. Liebowitz, and K. S. Pister, "Smart dust: Communicating with a cubic-millimeter computer," *Computer*, vol. 34, no. 1, pp. 44–51, 2001.

[6] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed, "Detection, classification, and tracking of targets," *IEEE signal processing magazine*, vol. 19, no. 2, pp. 17–29, 2002.

[7] A. Bates, M. L. Ling, J. Mann, and D. K. Arvind, "Respiratory rate and flow waveform estimation from tri-axial accelerometer data," in *Body Sensor Networks (BSN), 2010 International Conference on*, pp. 144–150, IEEE, 2010.

[8] "HOBNET FP7 Research Project." `http://www.hobnet-project.eu/files/D1.1.PDF`, Accessed 1-December-2012.

[9] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao, "Habitat monitoring: Application driver for wireless communications technology," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 2 supplement, pp. 20–41, 2001.

[10] F. Cagnacci, L. Boitani, R. A. Powell, and M. S. Boyce, "Animal ecology meets gps-based radiotelemetry: a perfect storm of opportunities and challenges," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 365, no. 1150, pp. 2157–2162, 2010.

[11] A. D. Young, M. J. Ling, and D. K. Arvind, "Distributed estimation of linear acceleration for improved accuracy in wireless inertial motion capture," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pp. 256–267, ACM, 2010.

[12] A. D. Young, "Use of body model constraints to improve accuracy of inertial motion capture," in *Body Sensor Networks (BSN), 2010 International Conference on*, pp. 180–186, IEEE, 2010.

[13] A. D. Young, "From posture to motion: The challenge for real time wireless inertial motion capture," in *Proceedings of the Fifth International Conference on Body Area Networks*, pp. 131–137, ACM, 2010.

[14] A. Markham, N. Trigoni, S. A. Ellwood, and D. W. Macdonald, "Revealing the hidden lives of underground animals using magneto-inductive tracking," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, pp. 281–294, ACM, 2010.

[15] R. Stolsmark and R. Tossebro, "Reducing energy consumption in a sheep tracking network using a cluster-based approach," 2012.

[16] R. Moen, J. Pastor, Y. Cohen, and C. C. Schwartz, "Effects of moose movement and habitat use on gps collar performance," *Journal of Wildlife Management*, vol. 60, pp. 659–668, July 1996.

[17] R. S. Rempel, A. R. Rodgers, and K. F. Abraham, "Performance of a gps animal location system under boreal forest canopy," *Journal of Wildlife Management*, vol. 59, pp. 543–551, July 1995.

[18] J. L. Bowman, C. O. Kochanny, S. Demarais, and B. D. Leopold, "Evaluation of gps collar for white tailed deer," *Wildlife Society Bulletin*, vol. 28, pp. 141–145, Spring 2000.

[19] C. R. Anderson and F. G. Lindzey, "Estimating cougar predation rates from gps location clusters," *Journal of Wildlife Management*, vol. 67, pp. 307–316, April 2003.

[20] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," in *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS X, pp. 96–107, ACM, 2002.

[21] D. Anthony, W. P. Bennett, M. C. Vuran, M. B. Dwyer, S. Elbaum, A. Lacy, M. Engels, and W. Wehtje, "Sensing through the continent: Towards monitoring migratory birds using cellular sensor networks," in *Proceedings of the 11th International Conference on Information Processing in Sensor Networks*, IPSN '12, pp. 329–340, ACM, 2012.

[22] C. Liu, B. Li, D. Fang, S. Guo, X. Chen, and T. Xing, "Demo: Rhinopithecus roxellana monitoring and identification using wireless sensor networks," in *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, SenSys '11, pp. 427–428, ACM, 2011.

[23] "Vectronic Aerospace - innovatice technology for science and research." `http://www.vectronic-aerospace.com`, Accessed 14-June-2015.

[24] "Vertex Survey Globalstar." Datasheet available at `http://www.vectronic-aerospace.com/files/VERTEX%20SURVEY%20Globalstar%20Collar%202.0.pdf`, Accessed 14-June-2015.

[25] "Vertex Survey Iridium." Datasheet available at `http://www.vectronic-aerospace.com/files/VERTEX%20SURVEY%20Iridium%20Collar.pdf`, Accessed 14-June-2015.

[26] I. E. Radoi, J. Mann, and D. Arvind, "Performance evaluation of the vb-tdma protocol for long-term tracking and monitoring of mobile entities in the outdoors," in *Proceedings of the 11th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, pp. 131–138, ACM, 2015.

[27] I. E. Radoi, J. Mann, and D. Arvind, "Tracking and monitoring horses in the wild using wireless sensor networks," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2015 IEEE 11th International Conference on*, pp. 732–739, IEEE, 2015.

[28] J. Mann, I. E. Radoi, and D. Arvind, "Prospeckz-5 - a wireless sensor platform for tracking and monitoring of wild horses," in *Digital System Design (DSD), 2014 17th Euromicro Conference on*, pp. 700–703, IEEE, 2014.

[29] D. Crisan, I. Radoi, and D. Arvind, "Coap-mediated hybrid simulation and visualisation environment for specknets," in *Proceedings of the 2013 ACM SIGSIM conference on Principles of advanced discrete simulation*, pp. 285–294, ACM, 2013.

[30] "Bud industries." Available at `www.budind.com`, Accessed 31-July-2016.

[31] I. Radoi, "Performance evaluation of routing protocols in lossy links for smart building networks," *MSc. Dissertation, The University of Edinburgh*, 2011.

[32] I. E. Radoi, A. Shenoy, and D. Arvind, "Evaluation of routing protocols for internet-enabled wireless sensor networks," in *Proceedings of the Eighth International Conference on Wireless and Mobile Communications, Venice, Italy*, pp. 24–29, IARIA, 2012.

[33] M. K. Marina, V. Radu, and K. Balampekos, "Impact of indoor-outdoor context on crowdsourcing based mobile coverage analysis," in *Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges*, pp. 45–50, ACM, 2015.

[34] V. Radu, P. Katsikouli, R. Sarkar, and M. K. Marina, "A semi-supervised learning approach for robust indoor-outdoor detection with smartphones," in *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pp. 280–294, ACM, 2014.

[35] V. Radu and M. K. Marina, "Himloc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced wifi fingerprinting," in *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, pp. 1–10, IEEE, 2013.

[36] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pp. 197–210, ACM, 2012.

[37] Y. Kim, H. Shin, Y. Chon, and H. Cha, "Smartphone-based wi-fi tracking system exploiting the rss peak to overcome the rss variance problem," *Pervasive and Mobile Computing*, vol. 9, no. 3, pp. 406–420, 2013.

[38] V. Radu, N. D. Lane, S. Bhattacharya, C. Mascolo, M. K. Marina, and F. Kawsar, "Towards multimodal deep learning for activity recognition on mobile devices," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pp. 185–188, ACM, 2016.

[39] I. Amundson and X. D. Koutsoukos, "A survey on localization for mobile wireless sensor networks," in *Mobile Entity Localization and Tracking in GPS-less Environnments*, pp. 235–254, Springer, 2009.

[40] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, C. Mascolo, B. Pasztor, S. Scellato, N. Trigoni, R. Wohlers, and K. Yousef, "Evolution and sustainability of a wildlife monitoring sensor network," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, pp. 127–140, ACM, 2010.

[41] M. Dessouky, R. Hall, A. Nowroozi, and K. Mourikas, "Bus dispatching at timed transfer transit stations using bus tracking technology," *Transportation Research Part C: Emerging Technologies*, vol. 7, no. 4, pp. 187–208, 1999.

[42] M. Kärkkäinen, "Increasing efficiency in the supply chain for short shelf life goods using rfid tagging," *International Journal of Retail & Distribution Management*, vol. 31, no. 10, pp. 529–536, 2003.

[43] T. Prusik, R. M. Arnold, and A. P. Piechowski, "Useful in tracking the thermal exposure history of a temperature-sensitive perishable product and providing a visually-distinct signal, such as a change in color density," Apr. 8 2003. US Patent 6,544,925.

[44] D. J. Montanari and G. M. Coleman, "Method for tracking the production history of food products," Dec. 26 1995. US Patent 5,478,990.

[45] M. S. Araki, P. Coe-Verbica, A. Banerjee, S. Shah, and M. Bajwa, "Animal-herd management using distributed sensor networks," April 2011.

[46] F. Dressler, S. Ripperger, M. Hierold, T. Nowak, C. Eibel, B. Cassens, F. Mayer, K. Meyer-Wegener, and A. Kolpin, "From radio telemetry to ultra-low-power sensor networks: tracking bats in the wild," *IEEE Communications Magazine*, vol. 54, no. 1, pp. 129–135, 2016.

[47] T. Nowak, A. Koelpin, F. Dressler, M. Hartmann, L. Patino, and J. Thielecke, "Combined localization and data transmission in energy-constrained wireless sensor networks," in *Wireless Sensors and Sensor Networks (WiSNet), 2015 IEEE Topical Conference on*, pp. 4–6, IEEE, 2015.

[48] M. Hartmann, T. Nowak, L. Patino-Studencki, J. Robert, A. Heuberger, and J. Thielecke, "A low-cost rssi-based localization system for wildlife tracking," in *IOP Conference Series: Materials Science and Engineering*, vol. 120, p. 012004, IOP Publishing, 2016.

[49] O. Osechas, J. Thiele, J. A. B. Link, and K. Wehrle, "Ratpack: Wearable sensor networks for animal observation," in *Proceedings of the IEEE 30th Annual International Conference on Engineering in Medicine and Biology*, EMBS 2008, pp. 538–541, IEEE, 2008.

[50] M. P. Johnson and P. L. Tyack, "A digital acoustic recording tag for measuring the response of wild marine mammals to sound," *IEEE Journal of Oceanic Engineering*, vol. 28, pp. 3–12, January 2003.

[51] M. Allen, L. Girod, R. Newton, S. Madden, D. T. Blumstein, and D. Estrin, "Voxnet: An interactive, rapidly-deployable acoustic monitoring platform," in *Proceedings of the 7th international conference on Information processing in sensor networks*, pp. 371–382, IEEE Computer Society, 2008.

[52] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, N. Trigoni, R. Wohlers, C. Mascolo, B. Pásztor, S. Scellato, and K. Yousef, "Wildsensing: Design and deployment of a sustainable sensor network for wildlife monitoring," *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, no. 4, p. 29, 2012.

[53] R. Kays, S. Tilak, M. Crofoot, T. Fountain, D. Obando, A. Ortega, F. Kuemmeth, J. Mandel, G. Swenson, T. Lambert, *et al.*, "Tracking animal location and

activity with an automated radio telemetry system in a tropical rainforest," *The Computer Journal*, p. bxr072, 2011.

[54] R. B. MacCurdy, R. M. Gabrielson, and K. A. Cortopassi, "Automated wildlife radio tracking," *Handbook of Position Location: Theory, Practice, and Advances*, pp. 1129–1167, 2011.

[55] A. C. Markham and A. J. Wilkinson, "Ecolocate: a heterogeneous wireless network system for wildlife tracking," in *Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics*, pp. 293–298, Springer, 2008.

[56] Š. Řeřucha, T. Bartonička, P. Jedlička, M. Čížek, O. Hlouša, R. Lučan, and I. Horáček, "The baara (biological automated radiotracking) system: a new approach in ecological field studies," *PloS one*, vol. 10, no. 2, p. e0116785, 2015.

[57] A. W. Weiser, Y. Orchan, R. Nathan, M. Charter, A. J. Weiss, and S. Toledo, "Characterizing the accuracy of a self-synchronized reverse-gps wildlife localization system," 2016.

[58] R. MacCurdy, R. Gabrielson, E. Spaulding, A. Purgue, K. Cortopassi, and K. Fristrup, "Automatic animal tracking using matched filters and time difference of arrival," *Journal of Communications*, vol. 4, no. 7, pp. 487–495, 2009.

[59] G. Strang and K. Borre, *Linear algebra, geodesy, and GPS*. Siam, 1997.

[60] G. C. White and R. A. Garrot, *Analysis of Wildlife Radio-Tracking Data*. an Diego, California, USA: Elsevier, 1990.

[61] M. R. Recio, R. Mathieu, P. Denys, P. Sirguey, and P. J. Seddon, "Lightweight gps-tags, one giant leap for wildlife tracking? an assessment approach," *PLoS One*, vol. 6, no. 12, p. e28225, 2011.

[62] "Fastrax IT430," October 2011. Datasheet available at `http://www.farnell.com/datasheets/1625182.pdf`, Accessed 31-July-2016.

[63] B. Cargnelutti, A. Coulon, A. M. Hewison, M. Goulard, J.-M. Angibault, and N. Morellet, "Testing global positioning system performance for wildlife monitoring using mobile collars and known reference points," *The Journal of Wildlife Management*, vol. 71, pp. 1380–1387, June 2007.

[64] B. P. Lo, S. Thiemjarus, R. King, and G.-Z. Yang, *Body sensor network–a wireless sensor platform for pervasive healthcare monitoring*. na, 2005.

[65] J. Hill and D. Culler, "Mica: a wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, no. 6, pp. 12–24, 2002.

[66] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pp. 364–369, IEEE, 2005.

[67] A. Milenković, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," *Computer communications*, vol. 29, no. 13, pp. 2521–2533, 2006.

[68] T. Choudhury, S. Consolvo, B. Harrison, J. Hightower, A. Lamarca, L. LeGrand, A. Rahimi, A. Rea, G. Bordello, B. Hemingway, *et al.*, "The mobile sensing platform: An embedded activity recognition system," *Pervasive Computing, IEEE*, vol. 7, no. 2, pp. 32–41, 2008.

[69] K.-J. Wong and D. K. Arvind, "Speckmac: Low-power decentralised mac protocols for low data rate transmissions in specknets," in *Proceedings of the 2Nd International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality*, REALMAN '06, pp. 71–78, ACM, 2006.

[70] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3 of *INFOCOM 2002*, pp. 1567–1576, IEEE, 2002.

[71] T. K. Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, pp. 171–180, ACM, 2003.

[72] L. van Hoesel and P. Havinga, "A lightweight medium access protocol (lmac) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches," 2004.

[73] J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks,” in *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys ’04, pp. 95–107, ACM, 2004.

[74] A. El-Hoiydi and J.-D. Decotignie, “Wisemac: An ultra low power mac protocol for multi-hop wireless sensor networks,” in *Algorithmic Aspects of Wireless Sensor Networks*, pp. 18–31, Springer, 2004.

[75] M. Buettner, G. Yee, E. Anderson, and R. Han, “X-mac: A short preamble mac protocol for duty-cycled wireless sensor networks,” in *Proceedings of the 4th international conference on Embedded Networked Sensor Systems*, pp. 307–320, ACM, 2006.

[76] K. Bilstrup, E. Uhlemann, E. Ström, and U. Bilstrup, “On the ability of the 802.11 p mac method and stdma to support real-time vehicle-to-vehicle communication,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, no. 1, pp. 1–13, 2009.

[77] F. Yu and S. Biswas, “A self reorganizing mac protocol for inter-vehicle data transfer applications in vehicular ad hoc networks,” in *Information Technology,(ICIT 2007). 10th International Conference on*, pp. 110–115, IEEE, 2007.

[78] Y. Bi, K.-H. Liu, L. X. Cai, X. Shen, and H. Zhao, “A multi-channel token ring protocol for qos provisioning in inter-vehicle communications,” *Wireless Communications, IEEE Transactions on*, vol. 8, no. 11, pp. 5621–5631, 2009.

[79] H. Su and X. Zhang, “Clustering-based multichannel mac protocols for qos provisionings over vehicular ad hoc networks,” *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3309–3323, 2007.

[80] N. Lu, Y. Ji, F. Liu, and X. Wang, “A dedicated multi-channel mac protocol design for vanet with adaptive broadcasting,” in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, pp. 1–6, IEEE, 2010.

[81] R. G. Bai, Y. G. Qu, Y. Guo, and B. H. Zhao, “An energy-efficient tdma mac for wireless sensor networks,” in *Asia-Pacific Service Computing Conference, The 2nd IEEE*, pp. 69–74, IEEE, 2007.

[82] M. S. Almalag, S. Olariu, and M. C. Weigle, “Tdma cluster-based mac for vanets (tc-mac),” in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pp. 1–6, IEEE, 2012.

[83] R. Fan, I. Chakraborty, and N. Lynch, “Clock synchronization for wireless networks,” in *Principles of distributed systems*, pp. 400–414, Springer, 2004.

[84] M. J. Booysen, S. Zeadally, and G.-J. van Rooyen, “Survey of media access control protocols for vehicular ad hoc networks,” *Communications, IET*, vol. 5, no. 11, pp. 1619–1631, 2011.

[85] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, “The evolution of mac protocols in wireless sensor networks: A survey,” *Communications Surveys and Tutorials, IEEE*, vol. 15, no. 1, pp. 101–120, 2013.

[86] Y. Kim, H. Shin, and H. Cha, “Y-mac: An energy-efficient multi-channel mac protocol for dense wireless sensor networks,” in *Proceedings of the 7th international conference on Information processing in sensor networks*, pp. 53–63, IEEE Computer Society, 2008.

[87] O. D. Incel, L. van Hoesel, P. Jansen, and P. Havinga, “Mc-lmac: A multichannel mac protocol for wireless sensor networks,” *Ad Hoc Networks*, vol. 9, no. 1, pp. 73–94, 2011.

[88] J. Borms, K. Steenhaut, and B. Lemmens, “Low-overhead dynamic multichannel mac for wireless sensor networks,” in *Wireless Sensor Networks*, pp. 81–96, Springer, 2010.

[89] G. P. Halkes and K. G. Langendoen, “Crankshaft: An energy-efficient mac-protocol for dense wireless sensor networks,” in *Proceedings of the 4th European Conference*, EWSN 2007, pp. 228–244, Springer Berlin Heidelberg, 2007.

[90] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, “Pw-mac: An energy-efficient predictive-wakeup mac protocol for wireless sensor networks,” in *INFOCOM, 2011 Proceedings IEEE*, pp. 1305–1313, IEEE, 2011.

[91] V. Shnayder, M. Hempstead, B. rong Chen, G. Werner-Allen, and M. Welsh, “Simulating the power consumption of large-scale sensor network applications,” *ACM Conference on Embedded Networked Systems (SenSys)*, 2004.

[92] C. Mallanda, A. Suri, V. Kunchakarra, S. Iyengar, R. Kannan, A. Durresi, and S. Sastry, "Simulating wireless sensor networks with omnet++," *Submitted to IEEE Computer*, 2005.

[93] "The network simulator - ns-2," September 2014. Available at `http://www.isi.edu/nsnam/ns/`, Accessed 31-July-2016.

[94] S. Park, A. Savvides, and M. B. Srivastava, "Sensorsim: A simulation framework for sensor networks," *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, 2000.

[95] A. Sobeih, W. Chen, J. C. Hou, L. Kung, N. Li, H. Lim, H. Tyan, and H. Zhang, "J-sim: a simulation and emulation environment for wireless sensor networks," *IEEE Wireless Communications Magazine*, vol. 13, pp. 104–119, November 2006.

[96] S. Park, A. Savvides, and M. B. Srivastava, "Simulating networks of wireless sensors," *The 2001 Winter Simulation Conference*, 2001.

[97] N. Aschenbruck, A. Munjal, and T. Camp, "Trace-based mobility modeling for multi-hop wireless networks," *Computer Communications*, vol. 34, no. 6, pp. 704–714, 2011.

[98] C. Tuduce and T. Gross, "A mobility model based on wlan traces and its validation," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 1, pp. 664–674, IEEE, 2005.

[99] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.

[100] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pp. 53–60, ACM, 1999.

[101] C. Bettstetter, "Smooth is better than sharp: A random mobility model for simulation of wireless networks," in *Proceedings of the 4th ACM international*

workshop on Modeling, analysis and simulation of wireless and mobile systems, pp. 19–27, ACM, 2001.

[102] W. jen Hsu, K. Merchant, H. wei Shu, C. hsin Hsu, and A. Helmy, “Weighted waypoint mobility model and its impact on ad hoc networks,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 1, pp. 59–63, 2005.

[103] M. Musolesi and C. Mascolo, “Mobility models for systems evaluation a survey,” *Middleware for Network Eccentric and Mobile Applications, Springer-Verlag*, 2008.

[104] F. Bai, N. Sadagopan, and A. Helmy, “Important: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks,” in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2, pp. 825–835, IEEE, 2003.

[105] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile computing*, pp. 153–181, Springer, 1996.

[106] “Sillicon labs efm32gg990.” Available at `https://www.silabs.com/Support%20Documents/TechnicalDocs/EFM32GG990.pdf`, Accessed 31-July-2016.

[107] “The haversine formula,” September 2014. Available at `https://en.wikipedia.org/wiki/Haversine_formula`, Accessed 31-July-2016.

[108] “Horse gait.” Available at `https://en.wikipedia.org/wiki/Horse_gait`, Accessed 31-July-2016.

[109] “Globaltop Technology Inc. FGPMMOPA6H,” 2011. Datasheet available at `http://www.adafruit.com/datasheets/GlobalTop-FGPMMOPA6H-Datasheet-V0A.pdf`, Accessed 31-July-2016.

[110] “Varta ER AA Lithium Thionyl Chloride.” Datasheet available at `http://www.varta-microbattery.com/applications/mb_data/documents/data_sheets/DS7106.pdf`, Accessed 31-July-2016.

[111] "ARM Cortex-M3 Processor." Description available at `http://www.arm.com/products/processors/cortex-m/cortex-m3.php`, Accessed 31-July-2016.

[112] PLANET FP7 Project, "Biological application document: Wildlife monitoring applications," 2013.

[113] J. Mann, R. Rabinovich, A. Bates, S. Giavedoni, W. MacNee, and D. K. Arvind, "Simultaneous activity and respiratory monitoring using an accelerometer," in *Proceedings of the 2011 International Conference on Body Sensor Networks*, BSN 2011, pp. 139–143, IEEE, 2011.

[114] "Pearson product-moment correlation coefficient." Description available at `https://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient`, Accessed 31-July-2016.

[115] G. R. Pedersen, E. Sondergaard, and J. Ladewig, "The influence of bedding on the time horses spend recumbent," *Journal of Equine Veterinary Science*, vol. 24, no. 4, pp. 153–158, 2004.

[116] L. E. Boyd, "The behaviour of przewalski's horses and its importance to their management," *Applied Animal Behaviour Science*, vol. 29, no. 1-4, pp. 301–318, 1991.

[117] J. D. Feist and D. R. McCullough, "Behavior patterns and communication in feral horses," *Zeitschrift für Tierpsychologie*, vol. 41, no. 4, pp. 337–371, 1976.

[118] N. Burri, P. Von Rickenbach, and R. Wattenhofer, "Dozer: ultra-low power data gathering in sensor networks," in *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, pp. 450–459, IEEE, 2007.

[119] M. Coyne and B. Godley, "Satellite tracking and analysis tool(stat): an integrated system for archiving, analyzing and mapping animal tracking data," *Marine Ecology Progress Series*, vol. 301, pp. 1–7, 2005.

[120] T. Ito, N. Miura, B. Lhagvasuren, D. Enkhbileg, S. Takatsuki, A. Tsunekawa, and Z. Jiang, "Satellite tracking of mongolian gazelles (procapra gutturosa) and habitat shifts in their seasonal ranges," *Journal of Zoology*, vol. 269, no. 3, pp. 291–298, 2006.

[121] V. Radu, L. Kriara, and M. K. Marina, "Pazl: A mobile crowdsensing based indoor wifi monitoring system," in *Network and Service Management (CNSM), 2013 9th International Conference on*, pp. 75–83, IEEE, 2013.

[122] V. Radu, J. Li, L. Kriara, M. K. Marina, and R. Mortier, "A hybrid approach for indoor mobile phone localization," *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys), ACM*, 2012.